



# Siebel Database Upgrade Guide

Version 7.7, Rev N  
March 2007

**ORACLE®**

Copyright © 2005, 2007, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

**PRODUCT MODULES AND OPTIONS.** This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

**U.S. GOVERNMENT RIGHTS.** Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

# Contents

## **Chapter 1: What's New in This Release**

## **Chapter 2: About Siebel Database Upgrade Topics**

- How the Siebel Database Upgrade Topics Are Organized 31
- About Applicability of Siebel Database Upgrade Topics 32
- About Terms Used in Siebel Database Upgrade Topics 33
- About File Paths and Commands in Siebel Database Upgrade Topics 34

## **Chapter 3: How to Perform a Siebel Database Upgrade**

- Road Map for Performing a Siebel Database Upgrade 37
- Process of Planning a Siebel Database Upgrade 39
- Process of Upgrading a Siebel Development Environment 40
- Process of Upgrading a Siebel Production Test Environment 47
- Process of Tuning Siebel Upgrade Performance 54
- Process of Upgrading a Siebel Production Environment 56

## **Chapter 4: How the Siebel Database Upgrade Works**

- About Supported Siebel Upgrade Paths 65
- About Siebel Upgrade Environments 66
- About the Siebel Database Upgrade Process 68
- About the Siebel Database Server Configuration Utilities 72
- About the Siebel Upgrade Wizard and Driver Files 77
- About the Siebel Database Upgrade Log Files 80
- ForExample of a Siebel Development Upgrade Flow 82
- About the Siebel Repository Merge 89
- About Inheriting Upgrade Behavior in a Siebel Upgrade 94
- About the Siebel Postmerge Utilities 97
- About the Siebel Incorporate Custom Layout (ICL) Upgrade Option 100

About the Siebel Database Server 109  
About Tuning Siebel Production Upgrade Files 109

## **Chapter 5: Siebel Database and UI Upgrade Planning**

About Siebel Upgrade Planning Resources 115  
About Best Practices for Doing Your Siebel Database Upgrade 118  
About New Siebel Upgrade Features and Requirements 121  
About Upgrading Your RDBMS in the Siebel Environment 121  
About Siebel User Interface Changes 122  
About Database Sort Order in the Siebel Environment 123  
About the Siebel Party Model 124  
About Migrating Siebel HTML Attachments to Base Tables 127  
About Siebel Database Growth Estimates 128  
Upgrade Planning for Multilingual Siebel Deployments 129  
Upgrade Planning for Siebel Unicode Support 130  
Upgrade Planning for Siebel RC2 or AES Encryption 132  
Upgrade Planning for Siebel Web Template Files and Style Sheet 132  
Upgrade Planning for Siebel Access Control 134  
Upgrade Planning for Siebel Database Clustered Indexes 135

## **Chapter 6: Application Planning for a Siebel Upgrade**

Determining Your Siebel 7.5.x Release Level 137  
Upgrade Planning for Migrating Siebel Address Data 139  
Upgrade Planning for Siebel Employee Relationship Management (ERM) 140  
Upgrade Planning for Siebel Marketing 141  
Upgrade Planning for Siebel Workflow Designer 147  
Upgrade Planning for Handheld Devices in the Siebel Environment 148  
Upgrade Planning for Resonate Central Dispatch in the Siebel Environment 149  
Upgrade Planning for Siebel String Translation 149  
Upgrade Planning for Siebel Configurator 149  
Upgrade Planning for Siebel Personalization 150  
Upgrade Planning: Additional Siebel Application Changes 150

## **Chapter 7: Basic Database Preparations for a Siebel Upgrade**

- Verifying Siebel Database Connectivity 151
- Preparing Siebel Tables and Views for Upgrade 151
- Preparing Siebel Custom Indexes for Upgrade 152
- Exporting Siebel Interface Table Data 153
- Archiving Unneeded Siebel Repositories 153
- Preserving Siebel Dock Objects and Visibility Rules 154
- Securing AIX Memory Allocation Segment Space for the Siebel Database 154

## **Chapter 8: Preparing an IBM DB2 Database for a Siebel Upgrade**

- Verifying the IBM DB2 Client for a Siebel Upgrade 157
- Verifying IBM DB2 Sort Order for a Siebel Upgrade 158
- Setting IBM DB2 Parameters for a Siebel Upgrade 158
- Verifying IBM DB2 Permissions for a Siebel Upgrade 160
- Verifying IBM DB2 Instance Owner Permissions for a Siebel Upgrade 161
- Creating IBM DB2 Temporary Tablespaces and Bufferpools for a Siebel Upgrade 161
- Analyzing IBM DB2 Custom Tablespace Requirements for a Siebel Upgrade 162
- Verifying the DB2 Application Development Client for a Siebel Upgrade 164
- Identifying IBM DB2 Long Columns for Truncation in a Siebel Upgrade 165
- Installing IBM DB2 Stored Procedures and Functions for a Siebel Upgrade 166

## **Chapter 9: Preparing an Oracle Database for a Siebel Upgrade**

- Verifying Oracle Database Sort Order for a Siebel Upgrade 169
- Setting Oracle Database Configuration Parameters for a Siebel Upgrade 170
- Verifying Oracle Database Parameters for Multiple CPUs in a Siebel Upgrade 172
- Verifying the Oracle Database ODBC Definition for a Siebel Upgrade 172

## **Chapter 10: Preparing an Microsoft SQL Server Database for a Siebel Upgrade**

- Verifying Microsoft SQL Server Sort Order for a Siebel Upgrade 175
- Setting Microsoft SQL Server Temporary Space Size for a Siebel Upgrade 176

Setting Microsoft SQL Server Configuration Parameters for a Siebel Upgrade 177  
Rebuilding Microsoft SQL Server Clustered Indexes for a Siebel Upgrade 178

## **Chapter 11: Preparing Siebel Application Data for Upgrade**

Preparing Siebel Workflow Processes for Upgrade 179  
Preparing Siebel Customized Seed Data for Upgrade 181  
Migrating Siebel Household Data 182  
Setting Up Campaign Status Values for Siebel Marketing 184  
Preserving Siebel Marketing Segment Descriptions 187  
Preparing Siebel Mobile User Data for Upgrade 187  
Setting the Value of Siebel S\_SRC\_PAYMENT.TYPE\_CD for Upgrade 188  
Preparing Siebel Address Data for Upgrade 189  
Migrating Siebel Address Data from Custom Extension Columns 190  
Preparing Siebel Customizable Product Data for Upgrade 197  
Preparing Siebel Configurator Data for Upgrade 197  
Resolving Duplicate Row IDs for a Siebel Upgrade 198  
Preparing Siebel eChannel Data for Upgrade 199  
Preparing Siebel Forecasting Data for Upgrade 199  
Preparing Siebel Products and Quotes Data for Upgrade 200  
Preparing Siebel Financial Services Tables for Upgrade 201

## **Chapter 12: Upgrading the Siebel Database**

Renaming the Siebel Repository 203  
Changing the Siebel Database Server Configuration Utilities Language 204  
Preparing to Run the Siebel Database Configuration Wizard 205  
Running the Siebel Database Configuration Wizard Under Windows 208  
Running the Siebel Database Configuration Wizard Under UNIX 209  
Starting the Siebel Upgrade Wizard 211  
Moving the Siebel Repository Files 214  
Regenerating SQL Files for a Siebel Upgrade 215  
Identifying and Dropping Obsolete Indexes for a Siebel Upgrade 215  
Preparing for a No-Development-Environment Siebel Upgrade 217

Installing New Siebel License Keys During an Upgrade 217

## **Chapter 13: Reviewing the Siebel Upgrade Log Files**

Summarizing Siebel Log Files Using the Logparse Utility 219

Reviewing Siebel Upgrade Log Files for Errors 221

Manually Archiving Siebel Upgrade Log Files 226

## **Chapter 14: Performing the Siebel Repository Merge**

Configuring Siebel Repository Objects to Inherit Upgrade Behavior 227

Configuring Siebel Tools for the Repository Merge 228

Performing a Siebel Repository Merge 230

Reviewing the Siebel Repository Merge Log Files 235

Generating Siebel EIM Temporary Columns 237

Reviewing Siebel Repository Object Property Conflicts 238

Regenerating the Siebel Repository Definition Files 240

Setting Label Alignment for Siebel Text Fields 242

Copying UI Files to a New Siebel Environment 243

Deleting Unneeded Siebel Repository Files 245

Running the Siebel Repository Preparation Wizard 245

## **Chapter 15: Migrating Siebel 6.x UI Customizations**

About Migrating to the Siebel Web Client 249

Identifying What Will Be Migrated in a Siebel 6.x Upgrade 254

Exposing Hidden Properties of Siebel Applets and Views 255

Running the Siebel Web Client Migration Wizard 256

Reviewing Siebel Applet and View Migration 259

Remigrating Siebel Applets and Views 260

Regenerating the Siebel Postmerge Utilities Log 261

## **Chapter 16: Reviewing the Siebel User Interface**

Troubleshooting Postmerge Siebel UI Problems 263

Verifying Siebel Business Address Applet Configuration 265

Reviewing Siebel Grid-Based Applets 265

Reviewing Siebel UI Navigation	267
Reviewing Siebel Multi-Value Group (MVG) Shuttle Applets	268
Revising Siebel UI Rich Text Controls	270
Reviewing New Siebel UI Aggregate Categories	272
Revising Siebel Visibility Filters to Display Correctly	273
Assigning a Category and Type to Siebel Chart Views	274
Assigning a Category and Type to Siebel Explorer Views	275
Setting Up Navigation to Inaccessible Siebel Detail Views	276
Eliminating Obsolete Siebel UI Fields	277
Reviewing Siebel UI Objects Affected by Incorporate Custom Layout	278
Reviewing Required Fields in the Siebel UI	279
Assigning an Item Identifier to Siebel Web Template Items	279

## **Chapter 17: Postmerge Development Tasks for Siebel 6.x Upgrades**

About Migrating Siebel 6.x Scripts	281
About Migrating Siebel 6.x Client-Side Interfaces	283
Reviewing Siebel Customized Business Components	283
Deleting Duplicate Siebel EIM Mappings	284
Migrating Siebel Scripts Attached to Controls	285
Migrating Siebel Scripts Attached to Applets	287
Migrating Siebel Business Component, Business Service, and Application Scripts	290
Migrating Outbound Siebel COM Interfaces	291
Migrating Inbound Siebel COM Interfaces	292
Setting Siebel Visibility Modes for Access Control	295

## **Chapter 18: Siebel Postmerge Development Tasks**

Reviewing Objects Deleted from the Siebel Repository	299
Reviewing Obsolete Objects in the Siebel Repository	300
Migrating Custom Siebel Workflows	301
Upgrading to the Siebel Symbolic String Model	302
Dropping IBM DB2 8-KB Tablespaces and Buffers After a Siebel Merge	303

Updating Siebel Enterprise Application Integration (EAI) 303

## **Chapter 19: Postupgrade Tasks for the Siebel Database and File System**

Reapplying Schema Customizations in the Siebel Database 305  
 Checking for Inactivated EIM Table Columns in the Siebel Database 308  
 Validating Dock Objects and Rule Definitions in the Siebel Database 308  
 Verifying an Upgraded Oracle RDBMS After a Siebel Upgrade 309  
 Setting Oracle Database Parameters After a Siebel Upgrade 311  
 Verifying S\_EVT\_ACT P1 Index on IBM DB2 After a Siebel Upgrade 311  
 Updating Siebel File System Attachments 311

## **Chapter 20: Postupgrade Tasks for Siebel Applications**

Generating Siebel Reporting Relationships 318  
 Setting Up Siebel Global Time Zone Support 319  
 Upgrading Siebel ERM Approval Business Process Workflows 320  
 Upgrading Siebel ERM Customized Microsite and Group News Pages 321  
 Migrating Course Duration Information for Siebel Training 323  
 Upgrading the Launch Field in Siebel Training LOV 323  
 Upgrading the Test Status in Siebel Training 324  
 Upgrading Responsibilities in Siebel Marketing 325  
 Reviewing Siebel Marketing Campaign Data 325  
 Reviewing Renamed Fields in Siebel Marketing 326  
 Displaying Regions in Siebel Marketing 326  
 Revising Program Flowchart Icons in Siebel Marketing 327  
 Setting Default Campaign Execution Options in Siebel Marketing 328  
 Upgrading Activity Plans for Programs and Campaigns in Siebel Marketing 328  
 Upgrading the Newsletter Offer Type in Siebel Marketing 329  
 Configuring Universal Inbox in Siebel Marketing 329  
 Upgrading Siebel Purchase Orders 330  
 Configuring Siebel Asset-Based Ordering 331  
 Reviewing Siebel Address Data 332

Configuring Siebel Products and Quotes Data	333
Upgrading Siebel Seeded Workflows	336
Upgrading Inbound Siebel Workflows	336
Universal Customer Master (UCM) Upgrade Steps After Repository Merge	337
Removing Siebel Call Center Duplicate Logins	340
Renaming the Siebel Life Sciences Configuration File	341

## **Chapter 21: Tuning the Siebel Upgrade Files**

Starting and Stopping Siebel Upgrade Tuner	343
Managing Parallel Threads Using Siebel Upgrade Tuner	345
Managing Zero-Row SQL Commands Using Siebel Upgrade Tuner	348
Transferring UNIX Files for Use by Siebel Upgrade Tuner	351
Rolling Back Siebel Upgrade Tuner Changes	354

## **Appendix A: Siebel Marketing Upgrade Reference**

Obsolete Siebel Marketing Data	357
Obsolete Business Objects in Siebel Marketing	363
Obsolete or Replaced Views in Siebel Marketing	364
Hidden Fields in Siebel Marketing	373

## **Appendix B: Tables Modified or Seeded During a Siebel Upgrade**

Tables Seeded as of Siebel 7.7	375
Tables Modified During Upgrade from Siebel Release 7.0.x	383
Tables Modified During Upgrade from Siebel Siebel 6.x	391

## **Appendix C: Upgrading to eConfigurator Version 7**

What's New in eConfigurator Version 7	405
Product Attributes and Classification System	409
Default Values	409
Customizable Products	409
Product Rule Manager	410
Declarative User Interface Customization	410
Multiple Instantiation	410
Browser-Based Configuration	411

Overview of Upgrading to eConfigurator Version 7	411
Upgrading from eConfigurator Version 6.x to eConfigurator Version 7	412
Approach 1: Using the Standard Upgrade Tool	412
Overview	412
Preupgrade Tasks	413
What the Upgrade Script Does	419
Postupgrade Tasks	424
Limitations	427
Case Study	428
Approach 2: Modifying Your Existing Implementation	438
Overview	438
Analyzing Your Release 6.x Models	438
Designing Your New Class System	439
Designing Dynamic Attributes	439
Assigning Products to Classes	439
Defining Customizable Products	439
Defining the UI	440
Defining Rules	440
Defining Resources, Linked Items, and Scripts	440
Defining Pricing	440
Upgrading from Release 5.x Product Configurator to eConfigurator Version 7	440

## Index



# 1

## What's New in This Release

### What's New in Siebel Database Upgrade Guide, Version 7.7, Rev. M

Table 1 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 1. New Features in Siebel Database Upgrade Guide, Version 7.7, Rev. N

Topic	Description
<a href="#">"Upgrading to eConfigurator Version 7" on page 405</a>	New appendix. Addresses upgrade issues related to eConfigurator, Version 7.

### What's New in Siebel Database Upgrade Guide, Version 7.7, Rev. M

Table 1 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 2. New Features in Siebel Database Upgrade Guide, Version 7.7, Rev. M

Topic	Description
Book Title	The book title is revised from <i>Upgrade Guide</i> to <i>Siebel Database Upgrade Guide</i> .
<a href="#">"Process of Upgrading a Siebel Development Environment" on page 40</a>	Revised topic. The following revisions were made to the process checklist: <ul style="list-style-type: none"><li>■ The Upgrade Wizard must be run after both the upprep and upgphys.</li><li>■ After the repository merge, delete unneeded repositories.</li><li>■ After the upgrade, install license keys.</li></ul>
<a href="#">"Process of Upgrading a Siebel Production Test Environment" on page 47</a>	Revised topic. The following revisions were made to the process checklist: <ul style="list-style-type: none"><li>■ Before performing the upgrade, move the custrep.dat and schema.ddl files to the production test environment.</li></ul>

## What's New in Upgrade Guide, Version 7.7, Rev. L

Table 3 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 3. New Features in Upgrade Guide, Version 7.7, Rev. L

Topic	Description
<a href="#">"About the Siebel Database Upgrade Process" on page 68</a>	Revised topic. Clarified that both the production environment upprep and upgphys are run in the production test environment and redirected to the production environment.
<a href="#">"About the Siebel Repository Merge" on page 89</a>	Revised topic. Topic rewritten and expanded to address customer feedback.
<a href="#">"About the Siebel Postmerge Utilities" on page 97</a>	Revised topic. Topic rewritten and expanded to address customer feedback.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised topic. Topic rewritten and expanded to address customer feedback.
<a href="#">"Process of Upgrading a Siebel Production Environment" on page 56</a>	Revised topic. Clarified that the production upgphys is run in the production test environment and redirected to the production environment.
<a href="#">"About Siebel Upgrade Planning Resources" on page 115</a>	Revised topic. Added <i>Data Model Reference</i> to list of publications in the Siebel deployment documentation suite.
<a href="#">"About Upgrading Your RDBMS in the Siebel Environment" on page 121</a>	Revised topic. Refer to Technical Note 640 before installing Siebel 7.7.x on Oracle 10g. Oracle 10g upgrades can be run in CBO mode.
<a href="#">"Determining Your Siebel 7.5.x Release Level" on page 137</a>	New topic. Determine your 7.5.x release-level before installing enterprise components.
<a href="#">"Upgrade Planning: Additional Siebel Application Changes" on page 150</a>	Revised topic. Technical Note 511 on SupportWeb lists obsolete objects in the Siebel Repository.
<a href="#">"Setting Oracle Database Configuration Parameters for a Siebel Upgrade" on page 170</a>	Revised topic. Set UNDO_MANAGEMENT = MANUAL before the repository merge for Oracle 9i and later.
<a href="#">"Preparing Siebel Mobile User Data for Upgrade" on page 187</a>	Revised topic. Before upgrade, check the Remote Status view and resolve any insert conflicts.
<a href="#">"Regenerating SQL Files for a Siebel Upgrade" on page 215</a>	New topic. Describes how to use the Database Server Configuration Utilities to regenerate SQL files.
<a href="#">"Configuring Siebel Tools for the Repository Merge" on page 228</a>	Revised topic. Clarified when to revise tools.cfg and added information on Oracle 10g support.
<a href="#">"Setting Label Alignment for Siebel Text Fields" on page 242</a>	Revised topic. In main.css make changes to both the .mceLabel and .mceLabel2 sections. Also revise printmain.css so that Quick Print and Print Preview display labels correctly.

Table 3. New Features in Upgrade Guide, Version 7.7, Rev. L

Topic	Description
"Creating a New SRF File"	Deleted topic. To create an SRF file, refer to <i>Developing and Deploying Siebel Business Applications</i> .
"Setting Oracle Database Parameters After a Siebel Upgrade" on page 311	New topic. Lists database parameters that must be set in init.ora after the upgrade.

### What's New in Upgrade Guide, Version 7.7, Rev. K

Table 4 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 4. New Features in Upgrade Guide, Version 7.7, Rev. K

Topic	Description
"About the Siebel Upgrade Wizard and Driver Files" on page 77	Revised topic. Topic rewritten to explain role and structure of driver files.
"ForExample of a Siebel Development Upgrade Flow" on page 82	New topic. Explains the steps in a typical development environment upgrade.
<ul style="list-style-type: none"> <li>■ "Process of Upgrading a Siebel Development Environment" on page 40</li> <li>■ "Process of Upgrading a Siebel Production Test Environment" on page 47</li> </ul>	Revised topic. You must set up migration of address data on custom extension columns after running the Database Server Configuration Utilities in upgrep mode but before starting the Siebel Upgrade Wizard.
"Upgrade Planning: Additional Siebel Application Changes" on page 150	New topic. Lists additional application changes that may affect upgrade planning.
"Archiving Unneeded Siebel Repositories" on page 153	New topic. Export all repositories except the Siebel Repository before doing an upgrep.
"Preparing Siebel Workflow Processes for Upgrade" on page 179	Revised topic. Tables S_WF_PROP_VAL and S_WF_STEP_INST must be purged before upgrade.
"Preparing Siebel Customized Seed Data for Upgrade" on page 181	New topic. Customized seed data is preserved during upgrade, even if doing so adversely affects application function. Prior to upgrade, determine whether to retain seed data customizations.
"Migrating Siebel Address Data from Custom Extension Columns" on page 190	Revised topic. Clarifies which steps to perform for a development environment upgrade versus a production test environment upgrade.
"Updating Siebel File System Attachments" on page 311	Revised topic. Run chng_file_sys.bat on each obsolete table.

## What's New in Upgrade Guide, Version 7.7, Rev. J

Table 5 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 5. New Features in Upgrade Guide, Version 7.7, Rev. J

Topic	Description
<a href="#">"About the Siebel Database Server Configuration Utilities" on page 72</a>	Revised topic. Expanded the description of utility modes. Describes how to do a production upgrade using the SQL from the production test environment.
Process of Upgrading a 6.x Development Environment	Obsolete topic. Now part of <a href="#">"Process of Upgrading a Siebel Development Environment" on page 40</a> .
<a href="#">"Process of Upgrading a Siebel Development Environment" on page 40</a>	Revised topic. Process tasks are organized into groups and placed in chapters. A process step now refers you to all the tasks in a chapter. Steps applicable only to 6.x upgrades are labeled.
<a href="#">"Process of Upgrading a Siebel Production Test Environment" on page 47</a>	Revised topic. Process tasks are organized into groups and placed in chapters. A process step now refers you to all the tasks in a chapter. Steps applicable only to 6.x upgrades are labeled.
<a href="#">"Process of Tuning Siebel Upgrade Performance" on page 54</a>	New topic. Provides a checklist for using Upgrade Tuner in the production test environment to tune upgrade performance.
<a href="#">"Process of Upgrading a Siebel Production Environment" on page 56</a>	Revised topic. Process tasks are organized into groups and placed in chapters. A process step now refers you to all the tasks in a chapter. Steps applicable only to 6.x upgrades are labeled.
<a href="#">"Upgrade Planning for Siebel Personalization" on page 150</a>	New topic. Primary user role is replaced by primary user responsibility at Release 7.7. To ensure applet visibility, review conditional expressions in the Administration-Personalization screen and revise conditional expressions as needed after the upgrade.
<a href="#">"Verifying Siebel Database Connectivity" on page 151</a>	New topic. From the production test environment, you must be able to connect to the Siebel Database in both the development environment and the production environment.
<a href="#">"Setting Oracle Database Configuration Parameters for a Siebel Upgrade" on page 170</a>	Revised topic. Revised setting for db_cache_size to 394264576.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised topic. Clarified how layouts are preserved in an ICL upgrade.
<a href="#">"About Tuning Siebel Production Upgrade Files" on page 109</a>	Revised topic. Clarified and expanded. Describes how Upgrade Tuner works, when to use Upgrade Tuner, and how it modifies driver files.

Table 5. New Features in Upgrade Guide, Version 7.7, Rev. J

Topic	Description
<a href="#">“Starting and Stopping Siebel Upgrade Tuner” on page 343</a>	Revised topic. Describes how to exit without saving changes.
<a href="#">“Managing Parallel Threads Using Siebel Upgrade Tuner” on page 345</a>	Revised topic. Clarified and expanded. Describes how to create, edit, and delete parallel threads for table and index creation.
<a href="#">“Managing Zero-Row SQL Commands Using Siebel Upgrade Tuner” on page 348</a>	Revised topic. Clarified and expanded. Describes how to tune an upgrade by deactivating zero-row SQL commands.
<a href="#">“Transferring UNIX Files for Use by Siebel Upgrade Tuner” on page 351</a>	Revised topic. Clarified and expanded. Describes the workflow for transferring UNIX upgrade files to and from a Windows host.
<a href="#">“Rolling Back Siebel Upgrade Tuner Changes” on page 354</a>	New topic. Describes how to roll back changes Upgrade Tuner makes to upgrade files.

## What's New in Upgrade Guide, Version 7.7, Rev. I

Table 6 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 6. New Features in Upgrade Guide, Version 7.7, Rev. I

Topic	Description
<a href="#">“About Siebel Upgrade Planning Resources” on page 115</a>	Revised Topic. Oracle’s Siebel SupportWeb now has an installation and upgrade portal page.
<a href="#">“Upgrade Planning for Multilingual Siebel Deployments” on page 129</a>	Revised topic. For multilingual deployments, you must manually import language-specific repository strings and seed data after upgrep.
<a href="#">“Upgrade Planning for Migrating Siebel Address Data” on page 139</a>	New topic. Explains revised address migration tasks.
<a href="#">“Preparing Siebel Address Data for Upgrade” on page 189</a>	New topic. Before doing the upgrade, you must run a script to identify address records with duplicate row IDs.
<a href="#">“Setting the Value of Siebel S_SRC_PAYMENT.TYPE_CD for Upgrade” on page 188</a>	New topic. S_SRC_PAYMENT.TYPE_CD is a required field. For records where this is null, set a value prior to upgrade.
<a href="#">“Migrating Siebel Address Data from Custom Extension Columns” on page 190</a>	Revised topic. Provides revised steps and SQL scripting to migrate addresses from custom extension columns.

Table 6. New Features in Upgrade Guide, Version 7.7, Rev. I

Topic	Description
<a href="#">“About the Siebel Incorporate Custom Layout (ICL) Upgrade Option” on page 100</a>	Revised topic: <ul style="list-style-type: none"> <li>■ In an ICL merge, new controls from the New Siebel Repository are not copied to applets.</li> <li>■ In parent list views, you must select a record in the parent list applet to display view tabs. This is true for both ICL and non-ICL upgrades.</li> <li>■ After an ICL upgrade of objects that use the Upgrade Ancestor property, the layout of descendant objects may differ from ancestor objects.</li> </ul>
<a href="#">“Reviewing Siebel Address Data” on page 332</a>	New topic. The upgrade generates a report that lists address records with duplicate address names. Review this report and resolve duplicate records.
<a href="#">“About a New Column Added to S_SRC_Payment”</a>	Deleted topic. Replaced with a task. See <a href="#">“Setting the Value of Siebel S_SRC_PAYMENT.TYPE_CD for Upgrade” on page 188.</a>

## What's New in Upgrade Guide, Version 7.7, Rev. H

Table 7 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 7. New Features in Upgrade Guide, Version 7.7, Rev. H

Topic	Description
<a href="#">“About Best Practices for Doing Your Siebel Database Upgrade” on page 118</a>	Revised topic. Added steps for determining your upgrade path.
<a href="#">“About the Siebel Party Model” on page 124</a>	Revised topic. Added description of how Siebel Financial Services 6.x household data is migrated to the party model at Release 7.0.x.
<a href="#">“Verifying the IBM DB2 Client for a Siebel Upgrade” on page 157</a>	New topic. The 64-bit IBM DB2 client is not supported on the Siebel Server.
<a href="#">“Setting IBM DB2 Parameters for a Siebel Upgrade” on page 158</a>	Revised topic. Topic lists only those DB2 parameters that must be reset specifically for upgrades.
<a href="#">“Migrating Siebel Household Data” on page 182</a>	Revised topic: <ul style="list-style-type: none"> <li>■ Clarified how to migrate Siebel Financial Services household data to the party model.</li> <li>■ Described how to retain the Siebel 6.x form of household associations.</li> </ul>

Table 7. New Features in Upgrade Guide, Version 7.7, Rev. H

Topic	Description
<a href="#">"Running the Siebel Database Configuration Wizard Under UNIX" on page 209</a>	Revised topic. Corrected and clarified procedure steps.
<a href="#">"Moving the Siebel Repository Files" on page 214</a>	New topic. Describes how to stop the Siebel Upgrade Wizard.
<a href="#">"Migrating Siebel Address Data from Custom Extension Columns" on page 190</a>	Revised topic. Added missing step. You must add new extension columns to the Siebel Repository.
<a href="#">"Summarizing Siebel Log Files Using the Logparse Utility" on page 219</a>	Revised topic. For steps executed in parallel, logparse adds to the process time only the duration of the longest step.
<a href="#">"Reviewing Siebel Upgrade Log Files for Errors" on page 221</a>	Revised topic. Corrected file names of logs that can be ignored.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised topic. Added recommendations on setting the Upgrade Ancestor property on applets.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised topic: <ul style="list-style-type: none"> <li>■ Added recommendations on using ICL with the Upgrade Ancestor property.</li> <li>■ In an ICL merge, new controls from the New Siebel Repository are not copied to applets.</li> <li>■ In parent list views, you must select a record in the parent list applet to display view tabs. This is true for both ICL and non-ICL upgrades.</li> </ul>
Upgrading the IBM DB2 Instance	Deleted topic.
"Safeguarding the New Custom Repository Export File"	Deleted topic. Replaced by step in <a href="#">"Setting Label Alignment for Siebel Text Fields" on page 242</a> .

## What's New in Upgrade Guide, Version 7.7, Rev. G

Table 8 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 8. New Features in Upgrade Guide, Version 7.7, Rev. G

Topic	Description
<a href="#">"Upgrade Planning for Siebel Web Template Files and Style Sheet" on page 132</a>	New topic. Guidelines for upgrading customized Web template files and style sheet file (main.css).
<a href="#">"Identifying and Dropping Obsolete Indexes for a Siebel Upgrade" on page 215</a>	New topic. Optional step during the production test environment upgrade for identifying and dropping indexes that may be obsolete.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised Topic. Provides guidelines for when to use ICL. Describes how Web template files and style sheet file (main.css) are handled during an ICL repository merge.
<a href="#">"Configuring Siebel Tools for the Repository Merge" on page 228</a>	New topic. Oracle only. Prior to the repository merge, you must set <code>SqlStyle = Oracle</code> in <code>tools.cfg</code> .
<a href="#">"Verifying Siebel Business Address Applet Configuration" on page 265</a>	New topic. For 6.x and 7.0.x Siebel Industry Application (SIA) upgrades, you must reconfigure the Business Address applet.
<a href="#">"Copying UI Files to a New Siebel Environment" on page 243</a>	New topic. Describes how to move customized Web template files and style sheet file (main.css) to the production test and production environments.
<a href="#">"Assigning an Item Identifier to Siebel Web Template Items" on page 279</a>	New topic. If you select ICL during the merge and choose the Labels On Top option, you must modify main.css.
<a href="#">"Updating Siebel File System Attachments" on page 311</a>	Revised topic. Corrected the description of arguments for <code>chg_file_sys.bat</code> .
<a href="#">"Upgrading Siebel ERM Customized Microsite and Group News Pages" on page 321</a>	Revised topic. Corrected error in procedure.
Chapter 16, <a href="#">"Reviewing the Siebel User Interface."</a>	All topics rewritten to improve clarity. Tools procedures added where needed. New topics added covering specific sections of postmerge utilities log.

## What's New in Upgrade Guide, Version 7.7, Rev. F

Table 9 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 9. New Features in Upgrade Guide, Version 7.7, Rev. F

Topic	Description
<a href="#">"Process of Upgrading a Siebel Development Environment" on page 40</a>	In Postupgrade Tasks for Environment Setup, the step about renaming the repository is not needed and has been removed.  In "Review Postmerge Utilities Log" section, added an optional step for compiling an .srf file to assist in resolving attribute conflicts.
<a href="#">"Assigning an Item Identifier to Siebel Web Template Items" on page 279</a>	New topic. You must revise main.css if you perform an ICL upgrade and choose Label on Top.

## What's New in Upgrade Guide, Version 7.7, Rev. E

Table 10 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 10. New Features in Upgrade Guide, Version 7.7, Rev. E

Topic	Description
<a href="#">"About Applicability of Siebel Database Upgrade Topics" on page 32</a>	Revised topic. Describes new, simplified topic applicability categories. The new categories have been applied to all the topics in the book.
<a href="#">"About Upgrading Your RDBMS in the Siebel Environment" on page 121</a>	Revised topic. Describes conditions that require an RDBMS upgrade before performing the Siebel Database upgrade.
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	Revised topic. Clarifies how ICL affects UI objects.
<a href="#">"Regenerating the Siebel Repository Definition Files" on page 240</a>	New topic. Describes how to regenerate schema.ddl and custrep.dat files. Required if you make repository changes after upgrading the development environment.
<a href="#">"Checking for Inactivated EIM Table Columns in the Siebel Database" on page 308</a>	New topic. Some columns in EIM tables were inactivated in Release 7.7.x. Verify that this does not adversely affect application function after the upgrade.
<a href="#">"Troubleshooting Postmerge Siebel UI Problems" on page 263</a>	Revised topic. If fields or controls do not display in an applet, look in Siebel Tools for a Web template of the same name, but appended with -Expanded.

Table 10. New Features in Upgrade Guide, Version 7.7, Rev. E

Topic	Description
<a href="#">“Preparing Siebel Financial Services Tables for Upgrade” on page 201</a>	Revised topic. Added that this task is required for upgrades from Siebel Consumer Goods 6.0.x but is otherwise optional.
“Roadmap for Performing an Upgrade from Siebel 6.x”	Deleted topic. Combined this topic with <a href="#">“Road Map for Performing a Siebel Database Upgrade” on page 37</a> .
“Roadmap for Performing an Upgrade from Siebel 7.x”	Deleted topic. Combined this topic with <a href="#">“Road Map for Performing a Siebel Database Upgrade” on page 37</a> .
“Roadmap for Performing an Upgrade Without a Development Environment”	Deleted topic. Combined this topic with <a href="#">“Road Map for Performing a Siebel Database Upgrade” on page 37</a> .
“Dropping Custom Temporary Tables”	Deleted topic. Replaced by <a href="#">“Preparing Siebel Tables and Views for Upgrade” on page 151</a> .
“Disabling Customized Database Triggers”	Deleted topic. Replaced by <a href="#">“Preparing Siebel Tables and Views for Upgrade” on page 151</a> .
“Dropping Customized Database Views”	Deleted topic. Replaced by <a href="#">“Preparing Siebel Tables and Views for Upgrade” on page 151</a> .
“Exporting Interface Table Data”	Deleted topic. Replaced by <a href="#">“Preparing Siebel Tables and Views for Upgrade” on page 151</a> .
“Setting IBM DB2 Upgrade-Specific Parameters”	Deleted topic. Combined this topic with <a href="#">“Setting IBM DB2 Parameters for a Siebel Upgrade” on page 158</a> .  Tables now provide both an upgrade setting and a production setting for parameters.
“Running IBM DB2 Statistics”	Deleted topic. Replaced by a step in the upgrade process topics in <a href="#">Chapter 3, “How to Perform a Siebel Database Upgrade.”</a>
“Increasing MS SQL Server File Size”	Deleted topic. Combined this topic with <a href="#">“Setting Microsoft SQL Server Configuration Parameters for a Siebel Upgrade” on page 177</a> .
“Running MS SQL Statistics”	Deleted topic. Replaced by a step in the upgrade process topics in <a href="#">Chapter 3, “How to Perform a Siebel Database Upgrade.”</a>
“Setting Up an Oracle Rollback Segment”	Deleted topic. Combined this topic with <a href="#">“Setting Oracle Database Configuration Parameters for a Siebel Upgrade” on page 170</a> .
“Setting Oracle Tablespace Size”	Deleted topic. Combined this topic with <a href="#">“Setting Oracle Database Configuration Parameters for a Siebel Upgrade” on page 170</a> .

Table 10. New Features in Upgrade Guide, Version 7.7, Rev. E

Topic	Description
"Running Oracle Analyze Command"	Deleted topic. Replaced by a step in the upgrade process topics in <a href="#">Chapter 3, "How to Perform a Siebel Database Upgrade."</a>
"Preparing Siebel Configurator Models for Upgrade"	Deleted topic. Redundant. See <a href="#">"Preparing Siebel Products and Quotes Data for Upgrade"</a> on page 200.
"About Migrating Applets and Views"	Deleted topic. Combined this topic with <a href="#">"About Migrating to the Siebel Web Client"</a> on page 249.
"About Model Applets"	Deleted topic. Combined this topic with <a href="#">"About Migrating to the Siebel Web Client"</a> on page 249.
"Migrating 6.x Customizations Using Batch Mode"	Deleted topic. Replaced by <a href="#">"Running the Siebel Web Client Migration Wizard"</a> on page 256.
"Reviewing New Applet Migration"	Deleted topic. Replaced by <a href="#">"Reviewing Siebel Applet and View Migration"</a> on page 259.
"Applying the Web Layout to a Set of Applets"	Deleted topic. Replaced by <a href="#">"Remigrating Siebel Applets and Views"</a> on page 260.
"Reviewing Modified Applet Migration"	Deleted topic. Replaced by <a href="#">"Reviewing Siebel Applet and View Migration"</a> on page 259.
"Reviewing View Migration"	Deleted topic. Replaced by <a href="#">"Reviewing Siebel Applet and View Migration"</a> on page 259.
"Applying the Web Layout to a Set of Views"	Deleted topic. Replaced by <a href="#">"Remigrating Siebel Applets and Views"</a> on page 260.
"Reviewing Modified View Migration"	Deleted topic. Replaced by <a href="#">"Reviewing Siebel Applet and View Migration"</a> on page 259.
"Migrating Custom Columns on Obsolete Tables"	Deleted topic. Combined this topic with <a href="#">"Reapplying Schema Customizations in the Siebel Database"</a> on page 305.
"Reviewing Data Warehouse Schema Changes in Siebel Marketing"	Deleted topic. This 6.x upgrade topic is obsolete due to changes in Siebel Marketing Server implementation in Release 7.7.
"About Siebel Tables Modified or Seeded During Upgrade"	Deleted topic. Repetitive.

## What's New in Upgrade Guide, Version 7.7, Rev. D

Table 11 lists changes in this version of the documentation to support Release 7.7.2 of the software.

Table 11. New Features in Upgrade Guide, Version 7.7, Rev. D

Topic	Description
<a href="#">"About the Siebel Incorporate Custom Layout (ICL) Upgrade Option" on page 100</a>	New topic. Feature introduced in Siebel 7.7.2.
<a href="#">"Troubleshooting Postmerge Siebel UI Problems" on page 263</a>	New topic. Contains steps for reviewing the UI after the repository merge. Includes steps specific to ICL.
<a href="#">"Reviewing Siebel Grid-Based Applets" on page 265</a>	Revised topic. Added information for upgrades using ICL.
<a href="#">"Upgrade Planning for Siebel Unicode Support" on page 130</a>	Revised topic. Migration to Unicode is recommended if you will use Siebel Email Response, stored correspondence, or similar functionality.
<a href="#">"Preparing for a No-Development-Environment Siebel Upgrade" on page 217</a>	Added steps to create a reference database, which is needed for the Prepare for Production step in the Database Server Configuration Utilities.
<a href="#">"Upgrade Planning for Siebel String Translation" on page 149</a>	New topic. Requests customers to contact Oracle's Siebel Technical Support or Oracle's Siebel Expert Services if they have modified string mapping or translations in previous releases.
<a href="#">"Reviewing String Translations"</a>	Deleted topic. Contact Oracle's Siebel Expert Services for help with string translation issues.
<a href="#">"Renaming Siebel Financial Services Account Objects"</a>	Deleted topic. See Siebel Technical Note 428 for information on migrating Siebel Financial Services data to Siebel 7.x.
<a href="#">"Upgrading the Oracle Database Schema on Windows"</a>	Deleted topic. Replaced by the following topics: <ul style="list-style-type: none"> <li>■ <a href="#">"Preparing to Run the Siebel Database Configuration Wizard" on page 205</a></li> <li>■ <a href="#">"Running the Siebel Database Configuration Wizard Under Windows" on page 208</a></li> </ul>
<a href="#">"Upgrading the Oracle Database Schema on UNIX"</a>	Deleted topic. Replaced by the following topics: <ul style="list-style-type: none"> <li>■ <a href="#">"Preparing to Run the Siebel Database Configuration Wizard" on page 205</a></li> <li>■ <a href="#">"Running the Siebel Database Configuration Wizard Under UNIX" on page 209</a></li> </ul>

Table 11. New Features in Upgrade Guide, Version 7.7, Rev. D

Topic	Description
"Upgrading the Database Schema for MS SQL Server"	Deleted topic. Replaced by the following topics: <ul style="list-style-type: none"> <li>■ "Preparing to Run the Siebel Database Configuration Wizard" on page 205</li> <li>■ "Running the Siebel Database Configuration Wizard Under Windows" on page 208</li> </ul>
"Upgrading IBM DB2 on Windows"	Deleted topic. Replaced by the following topics: <ul style="list-style-type: none"> <li>■ "Preparing to Run the Siebel Database Configuration Wizard" on page 205</li> <li>■ "Running the Siebel Database Configuration Wizard Under Windows" on page 208</li> </ul>
"Upgrading IBM DB2 UDB on UNIX"	Deleted topic. Replaced by the following topics: <ul style="list-style-type: none"> <li>■ "Preparing to Run the Siebel Database Configuration Wizard" on page 205</li> <li>■ "Running the Siebel Database Configuration Wizard Under UNIX" on page 209</li> </ul>

## What's New in Upgrade Guide, Version 7.7, Rev. C

Table 12 lists changes in this version of the documentation to support Release 7.7 of the software.

Table 12. New Features in Upgrade Guide, Version 7.7, Rev. C

Topic	Description
"About Supported Siebel Upgrade Paths" on page 65	New topic. Defines the Siebel environments. All topics now indicate the upgrade path, environment, or database to which they apply.
"About the Siebel Database Upgrade Process" on page 68	New topic. Explains the key steps in the upgrade process.
"About the Siebel Database Server Configuration Utilities" on page 72	Revised topic. Explains the role of the Database Server Configuration Utilities.
"About the Siebel Upgrade Wizard and Driver Files" on page 77	Revised topic. Explains the role of the Siebel Upgrade Wizard.
"About Applicability of Siebel Database Upgrade Topics" on page 32	New topic. Explains the meaning of applicability statements. These statements have been added to the beginning of all task and process topics in the book.
Chapter 3, "How to Perform a Siebel Database Upgrade"	New chapter. Contains road map topics and process topics you can use as checklists for completing an upgrade.

Table 12. New Features in Upgrade Guide, Version 7.7, Rev. C

Topic	Description
"About Upgrading Your RDBMS in the Siebel Environment" on page 121	New topic. States that an RDBMS upgrade can be done before or after upgrading to the new Siebel release.
"Preparing for a No-Development-Environment Siebel Upgrade" on page 217	New topic. Explains how to move and rename repository and schema definition files for a no-development-environment upgrade.
"Verifying the Oracle Database ODBC Definition for a Siebel Upgrade" on page 172	Corrected the topic to explain that manually created Oracle 9i ODBC definitions must be edited after creating them.
"Reviewing Siebel Upgrade Log Files for Errors" on page 221	Corrected the procedure. Added a reference to <i>Troubleshooting Steps 21</i> on Oracle's Siebel SupportWeb.
"Verifying an Upgraded Oracle RDBMS After a Siebel Upgrade" on page 309	New topic. Describes how to use ddlimp to verify the schema after upgrading an Oracle RDBMS.
"Upgrade Planning for Siebel RC2 or AES Encryption" on page 132	Revised topic to remove procedures. Topic refers users to the <i>Security Guide for Siebel Business Applications</i> and to a Technical Note on security implementation.
"How the Siebel Database Upgrade Topics Are Organized" on page 31	New topic. Explains the new organization of the book.
"Increasing IBM DB2 ODBC Statement Handles"	Deleted topic. Replaced with a process task that requests the user to review guidelines for configuring the RDBMS in the <i>Siebel Installation Guide</i> for the operating system you are using.
Chapter 12, "Upgrading the Production Environment Clients"	Deleted topics on how to use Siebel Anywhere. For information on creating distribution kits, see <i>Siebel Anywhere Administration Guide</i> .
"Creating the BASE Package"	Deleted from Chapter 12, "Upgrading the Production Environment Clients." Topic will appear in the next revision of <i>Going Live with Siebel Business Applications</i> .
"Creating the Language Package"	Deleted from Chapter 12, "Upgrading the Production Environment Clients." Topic will appear in the next revision of <i>Going Live with Siebel Business Applications</i> .
"Deploying Workflows"	Deleted topic. For information on deploying workflows, see <i>Siebel Business Process Designer Administration Guide</i> .

Table 12. New Features in Upgrade Guide, Version 7.7, Rev. C

Topic	Description
Appendix A, "Upgrading Without a Development Environment"	Deleted. Replaced by road map topic in <a href="#">Chapter 3, "How to Perform a Siebel Database Upgrade."</a>
Reorganized the book.	Reorganized book to clarify flow of upgrade process and improve access to topics in the book. Use the road map and process topics in <a href="#">Chapter 3, "How to Perform a Siebel Database Upgrade"</a> to guide you through performing an upgrade.  All processes now assume the presence of a production test environment.

### What's New in Upgrade Guide, Version 7.7, Rev. B

[Table 13](#) lists changes in this version of the documentation to support Release 7.7 of the software.

Table 13. New Features in Upgrade Guide, Version 7.7, Rev. B

Topic	Description
<a href="#">"Setting Label Alignment for Siebel Text Fields" on page 242</a>	Added a requirement to copy schema.ddl to the production environment.
<a href="#">"About Siebel Database Growth Estimates" on page 128</a>	Corrected growth estimates for IBM DB2 and MSSQL databases.
<a href="#">"Migrating Siebel Address Data from Custom Extension Columns" on page 190</a>	Added a detailed description of the upgrade paths to which this topic applies.
<a href="#">"About the Siebel Repository Merge" on page 89</a>	New topic. Describes how repository merge works.
<a href="#">"About Inheriting Upgrade Behavior in a Siebel Upgrade" on page 94</a>	New topic. Describes how Upgrade Ancestor field works in a repository merge.
<a href="#">"About the Siebel Postmerge Utilities" on page 97</a>	New topic. Describes the role of postmerge utilities in repository merge.
<a href="#">"Reviewing the Siebel Repository Merge Log Files" on page 235</a>	Corrected procedural steps.
<a href="#">"Reviewing Siebel Repository Object Property Conflicts" on page 238</a>	Corrected procedural steps.
<a href="#">"Reviewing Objects Deleted from the Siebel Repository" on page 299</a>	New topic.
<a href="#">"Reviewing Obsolete Objects in the Siebel Repository" on page 300</a>	New topic.
<a href="#">"Creating a New SRF File"</a>	Corrected procedural steps.

Table 13. New Features in Upgrade Guide, Version 7.7, Rev. B

Topic	Description
"Upgrading Without a Development Environment"	Corrected procedural steps for creating custrep.dat and schema.ddl.
Incorporate Custom Layout (ICL)	Removed content on Incorporate Custom Layout. This feature is not supported at this time.

## What's New in Upgrade Guide, Version 7.7, Rev. A

Table 14 lists changes in this version of the documentation to support Release 7.7 of the software.

Table 14. New Features in Upgrade Guide, Version 7.7, Rev. A

Topic	Description
"Summarizing Siebel Log Files Using the Logparse Utility" on page 219	Corrected the command syntax for starting the logparse utility.
"Migrating Siebel Address Data from Custom Extension Columns" on page 190	Added a prerequisite stating that you must run the Database Server Configuration Utilities before performing the procedure.
"Setting Oracle Database Configuration Parameters for a Siebel Upgrade" on page 170	UNIX only. Added a prerequisite stating that if you have installed the Oracle 64-bit client, you must revise the LD_LIBRARY path.
"DB2 Database Manager Settings" on page 159	Corrected the 16-bit and 32-bit STAT_HEAP_SZ requirements.
"About File Paths and Commands in Siebel Database Upgrade Topics" on page 34	Corrected commands throughout the book to reflect the correct definition of SIEBEL_ROOT.
New book: <i>Siebel Database Upgrade Guide</i>	Combined <i>Upgrade Guide for Windows</i> and <i>Upgrade Guide for UNIX</i> into one book.

## What's New in Upgrade Guide, Version 7.7

Table 15 lists changes in this version of the documentation to support Release 7.7 of the software.

Table 15. New Features in Upgrade Guide, Version 7.7

Topic	Description
<a href="#">Chapter 11, "Preparing Siebel Application Data for Upgrade"</a>	This chapter describes new tasks for upgrades of specific Siebel modules, including Marketing, Employee Relationship Management (ERM), and Workflow Designer.
<a href="#">"Configuring Siebel Repository Objects to Inherit Upgrade Behavior" on page 227</a>	This topic describes new Siebel Repository merge features.
<a href="#">"Verifying the User Interface Configuration"</a>	<p>A new Siebel Repository merge option, Incorporate Custom Layouts (ICL), helps preserve view and applet customizations for 7.x upgrades.</p> <p>A new user interface navigation scheme is introduced in Release 7.7. New Siebel Repository postmerge utilities check customized user interface objects for common configuration problems. They also convert multi-value group applets to shuttle applets.</p>
<a href="#">Chapter 19, "Postupgrade Tasks for the Siebel Database and File System"</a>	This chapter includes new tasks for upgrading specific Siebel modules, including Marketing, Employee Relationship Management (ERM), and Workflow Designer.
<a href="#">"Transferring UNIX Files for Use by Siebel Upgrade Tuner" on page 351</a>	This section describes a new upgrade process to tune the production upgrade for improved performance.



# 2

## About Siebel Database Upgrade Topics

This chapter contains the following topics:

- [“How the Siebel Database Upgrade Topics Are Organized”](#) on page 31
- [“About Applicability of Siebel Database Upgrade Topics”](#) on page 32
- [“About Terms Used in Siebel Database Upgrade Topics”](#) on page 33
- [“About File Paths and Commands in Siebel Database Upgrade Topics”](#) on page 34

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## How the Siebel Database Upgrade Topics Are Organized

**Upgrades from:** All Siebel releases.

Use the road map and process topics in [Chapter 3, “How to Perform a Siebel Database Upgrade”](#) to guide you through the upgrade process. These topics provide a checklist of the steps required to complete an upgrade, in the order you must perform them. Each step includes a link to a topic that explains how to complete the step. The remaining chapters of the *Siebel Database Upgrade Guide* are organized according to the major phases of the upgrade.

**NOTE:** Topics in the chapters may not follow the order you perform them during the upgrade.

The order of topics in a chapter is as follows:

- A list of all the topics in the chapter.
- **Road map topic.** A road map is a numbered list of processes. For example, “Road Map for Performing the Upgrade” lists all the processes required to upgrade your Siebel environments.
- **Process topics.** A process topic has a numbered list of tasks. For example, “Process of Upgrading a Development Environment” lists the tasks required to upgrade a single Siebel environment. These topics begin with “Process of...”.
- **Concept topics.** Concept topics explain key concepts required to perform groups of tasks. To get an overview of how each part of the upgrade works, read the concept topics. These topics usually begin with “About...”. Concept topics do not begin with a gerund, “Process of,” or “Road Map for.”
- **Task topics.** Task topics explain how to do each step in an upgrade process, and typically include a procedure. Task topics begin with a gerund, for example, “Updating File System Attachments.”

### Related Topics

[“About Applicability of Siebel Database Upgrade Topics” on page 32](#)

[“About Terms Used in Siebel Database Upgrade Topics” on page 33](#)

[“About File Paths and Commands in Siebel Database Upgrade Topics” on page 34](#)

## About Applicability of Siebel Database Upgrade Topics

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

The applicability of topics in the book is listed at the beginning of each topic. [Table 16](#) lists the applicability categories and their meaning.

For each topic, only the relevant categories are listed.

Table 16. Topic Applicability Categories

Applicability Category	Meaning
Upgrades	Lists the upgrades to which the topic applies, for example: <b>Upgrades from:</b> Siebel 7.0.x, 7.5.x, & 7.7.x.
Environments	Lists the Siebel environments to which the topic applies.  For example, “ <b>Environments:</b> Development environment only” means the topic applies only to a development environment upgrade.
Databases	Lists the databases to which the topic applies.  For example, “ <b>Databases:</b> IBM DB2 only” means the topic applies only to IBM DB2 databases.
Platforms	Lists the operating system platform to which the topic applies.  For example, “ <b>Platforms:</b> Windows only” means the topic applies only to Windows platforms.  Windows and UNIX refer to the Enterprise Server platforms (one of which is used by all customers). IBM z/OS refers to the database platform also used by IBM DB2 for z/OS customers.

## Related Topics

[“About Siebel Upgrade Environments” on page 66](#)

[“How the Siebel Database Upgrade Topics Are Organized” on page 31](#)

[“About Terms Used in Siebel Database Upgrade Topics” on page 33](#)

[“About File Paths and Commands in Siebel Database Upgrade Topics” on page 34](#)

# About Terms Used in Siebel Database Upgrade Topics

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

This guide follows several naming conventions:

- Current release means the currently shipping release of the Siebel Business Applications.
- Siebel 8.x refers collectively to all versions of Siebel 8, for example, Release 8.0. See the Upgrade topic in *System Requirements and Supported Platforms* on Siebel SupportWeb for specific information on which release numbers are meant by Siebel 8.x.
- Siebel 7.x refers collectively to all versions of Siebel 7, for example, Release 7.0.4 or 7.5.3. See the Upgrade topic in *System Requirements and Supported Platforms* on Siebel SupportWeb for specific information on which release numbers are meant by Siebel 7.x.
- Siebel 7.0.x refers to versions of Siebel 7 prior to Release 7.5 that are supported for upgrade to the current release, for example, Release 7.0.4. See the Upgrade topic in *System Requirements and Supported Platforms* on Siebel SupportWeb for specific information on which release numbers are meant by Release 7.0.x.
- Siebel 6.x refers collectively to all versions of Siebel 2000 that are supported for upgrade to the current release. See the Upgrade topic in *System Requirements and Supported Platforms* on Siebel SupportWeb for specific information on which release numbers are meant by Siebel 6.x.
- The term “Windows” refers to all Microsoft Windows operating systems listed as supported for this release in *System Requirements and Supported Platforms* on Siebel SupportWeb.
- The term “UNIX” refers to all forms of the UNIX operating system supported for this release in *System Requirements and Supported Platforms* on Siebel SupportWeb.
- The term *IBM z/OS* refers to the IBM mainframe operating systems collectively referred as z/OS and supported for this release in *System Requirements and Supported Platforms* on Siebel SupportWeb.

## Related Topics

[“How the Siebel Database Upgrade Topics Are Organized” on page 31](#)

[“About Applicability of Siebel Database Upgrade Topics” on page 32](#)

[“About File Paths and Commands in Siebel Database Upgrade Topics” on page 34](#)

# About File Paths and Commands in Siebel Database Upgrade Topics

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

Environment variables and path placeholders for both Windows and UNIX paths are used throughout the *Siebel Database Upgrade Guide*. You should enter UNIX commands in a Korn shell. Enter Windows commands in a Windows Command Prompt window.

## Windows Paths

The following path conventions specify file system locations in *Siebel Database Upgrade Guide* topics:

- **SIEBEL\_ROOT** is the absolute path of the Siebel Server installation directory. When you install a Siebel Server, the installation program queries for the path to the installation directory. The installation program then installs the Siebel Server in a subdirectory of this path called siebsrvr. For example, if you specified C: \sea7xx as the installation directory for Siebel 7.x, then **SIEBEL\_ROOT** is C: \sea7xx\si ebsrvr.
- **DBSRVR\_ROOT** is the absolute path to the Siebel Database Server files on the Siebel Server. When you install the Siebel Database Server, the installation script queries for the path to the Siebel Server installation directory. The script then installs the Siebel Database Server files at the same level in a subdirectory called dbsrvr. For example, if **SIEBEL\_ROOT** is C: \sea7xx\si ebsrvr, then **DBSRVR\_ROOT** is C: \sea7xx\dbsrvr.
- Throughout the Guide, examples use the path C: \ and C: \sea7xx.

## UNIX Paths

The following environment variables and path conventions specify file system locations in *Siebel Database Upgrade Guide* topics:

- **SIEBEL\_ROOT** is an environment variable that defines the absolute path of the Siebel Server installation directory. When you install a Siebel Server, the installation script queries for the path to the installation directory. The script then installs the Siebel Server in a subdirectory of this path called siebsrvr. For example, if you specified /usr/siebel as the installation directory for Siebel 7.x, then **\$SIEBEL\_ROOT** is /usr/siebel/sea7xx/siebsrvr.
- The definition of **SIEBEL\_ROOT** and other environment variables required for doing an upgrade are located in /siebsrvr/siebenv.sh. The Siebel Server installation script sets environment variable definitions in this shell script. Do not edit or delete this file.

**TIP:** Before performing command line procedures, source siebenv.csh first. This refreshes the environment variables required to run commands.

- *DBSRVR\_ROOT* is a path convention used in the *Siebel Database Upgrade Guide*. It is not an environment variable and is not defined in `si ebenv.csh` or `si ebenv.sh`.

*DBSRVR\_ROOT* is the absolute path to the Siebel Database Server files on the Siebel Server. When you install the Siebel Database Server, the installation script queries for the Siebel Server installation directory. The script then installs the Siebel Database Server files at the same level in a subdirectory called `dbsrvr`. For example, if `$SIEBEL_ROOT` is `/usr/siebel/sea7xx/siebsrvr`, then *DBSRVR\_ROOT* is `/usr/siebel/sea7xx/dbsrvr`.

- Throughout the Guide, examples use the path `/usr/siebel` and `/usr/siebel/sea7xx`.
- Run UNIX scripts in a C or Korn shell. Do not run `.ksh` scripts in a Bourne shell.

## Path Navigation

Procedural steps that ask you to navigate to a specified directory should be performed as follows:

- Windows: Open a Command Prompt window and use the `cd` command to make the specified directory the current directory. Do not use the Windows File Explorer to navigate to the directory. For help with the `cd` command, enter the word `help` in the Command Prompt window and click ENTER.
- UNIX: In a shell window, make the specified directory the current directory.

## Executing Commands

Procedural steps that ask you to execute a command should be performed as follows, unless specified otherwise:

- Windows: In a Command Prompt window, verify the current directory is correct and enter the command. Do not run the command by entering it in the Run window in the Start Menu.
- UNIX: In a shell window, verify the current directory is correct, source the `siebenv` script, then enter the command.

Because all versions of the UNIX operating system are case-sensitive, if you are running your Siebel Business Applications under UNIX, treat all file names, directory names, path names, parameters, flags, and command-line commands as lowercase, unless you are instructed otherwise in the product.

If your deployment currently runs under Windows, but you might switch to a UNIX environment or deploy UNIX servers in the future, follow this same practice to avoid having to rename everything later.

## Related Topics

[“How the Siebel Database Upgrade Topics Are Organized” on page 31](#)

[“About Applicability of Siebel Database Upgrade Topics” on page 32](#)

[“About Terms Used in Siebel Database Upgrade Topics” on page 33](#)



# 3

## How to Perform a Siebel Database Upgrade

This chapter contains the following topics:

- [“Road Map for Performing a Siebel Database Upgrade” on page 37](#)
- [“Process of Planning a Siebel Database Upgrade” on page 39](#)
- [“Process of Upgrading a Siebel Development Environment” on page 40](#)
- [“Process of Upgrading a Siebel Production Test Environment” on page 47](#)
- [“Process of Tuning Siebel Upgrade Performance” on page 54](#)
- [“Process of Upgrading a Siebel Production Environment” on page 56](#)

### Related Topics

[Chapter 2, “About Siebel Database Upgrade Topics”](#)

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

## Road Map for Performing a Siebel Database Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

Use one of the road maps listed below to guide you through the steps for upgrading your Siebel database.

Each road map consists of a group of processes. Each process consists of a numbered list of tasks. After you complete the tasks in a process, go on to the next process in the road map. When you have completed all the processes in the road map, the upgrade is complete.

Depending on your database and installed release, you may not need to complete all the tasks in a process. Before starting a task, check the applicability information at the beginning of the task and verify the task applies to your upgrade.

Choose one of the following road maps to perform the upgrade:

- [“Road Map for Upgrading from Siebel 6.x” on page 38](#)
- [“Road Map for Upgrading from Siebel 7.x on IBM DB2 UDB for z/OS” on page 38](#)
- [“Road Map for Upgrading If You Have a Development Environment” on page 38](#)
- [“Road Map for Upgrading Without a Development Environment” on page 38](#)

## Road Map for Upgrading from Siebel 6.x

If you are upgrading from Siebel 6.x, you must first upgrade to Siebel 7.7. Then, you upgrade from Siebel 7.7 to the current release. To upgrade from Siebel 6.x to Siebel 7.7, see the latest Siebel 7.7 version of the *Siebel Database Upgrade Guide* on Siebel SupportWeb.

Direct upgrades from Siebel 6.2.1 of Siebel Financial Services to the current release of Siebel Industry Applications on IBM z/OS platforms are supported. To perform this upgrade, see *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

## Road Map for Upgrading from Siebel 7.x on IBM DB2 UDB for z/OS

Upgrading from Siebel 7.x on IBM DB2 UDB for z/OS is described in *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

The *Siebel Database Upgrade Guide for DB2 UDB for z/OS* contains steps that refer you to the present guide (*Siebel Database Upgrade Guide*) to perform certain tasks. A statement at the beginning of each of these tasks indicates that the task applies to IBM z/OS platforms. For example:

**Platforms:** MS Windows, UNIX, IBM z/OS.

Do not perform a task unless IBM z/OS is listed as an applicable platform.

After you have completed a task, refer to the *Siebel Database Upgrade Guide for DB2 UDB for z/OS* to perform the next upgrade step.

## Road Map for Upgrading If You Have a Development Environment

If you are upgrading and have a development environment, complete the processes in this road map in the order shown:

- 1 "Process of Planning a Siebel Database Upgrade" on page 39.
- 2 "Process of Upgrading a Siebel Development Environment" on page 40.
- 3 "Process of Upgrading a Siebel Production Test Environment" on page 47.
- 4 "Process of Tuning Siebel Upgrade Performance" on page 54.
- 5 "Process of Upgrading a Siebel Production Environment" on page 56.

## Road Map for Upgrading Without a Development Environment

If you are upgrading and do not have a development environment, complete the processes in this road map in the order shown.

Upgrading without a development environment means the following are true:

- You are running an uncustomized, out-of-the box version of Siebel applications.
- You have not used Siebel Tools to create or modify any objects or logical schema definitions in the Siebel Repository.
- You have not modified the physical schema in the Siebel database.

If your upgrade meets these criteria, complete the following processes in the order shown:

- 1 ["Process of Planning a Siebel Database Upgrade" on page 39.](#)
- 2 ["Preparing for a No-Development-Environment Siebel Upgrade" on page 217.](#)
- 3 Perform a production test environment upgrade. See ["Process of Upgrading a Siebel Production Test Environment" on page 47.](#)
- 4 ["Process of Tuning Siebel Upgrade Performance" on page 54.](#)
- 5 Upgrade your production environment. See ["Process of Upgrading a Siebel Production Environment" on page 56.](#)

## Process of Planning a Siebel Database Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** Windows and UNIX only. This topic does not apply to IBM z/OS.

This process is part of a road map. See ["Road Map for Performing a Siebel Database Upgrade" on page 37.](#)

To plan the upgrade, read the following:

- 1 [Chapter 4, "How the Siebel Database Upgrade Works."](#)
- 2 [Chapter 5, "Siebel Database and UI Upgrade Planning."](#)
- 3 If you are planning to migrate to Unicode, see ["Upgrade Planning for Siebel Unicode Support" on page 130.](#)
- 4 [Chapter 6, "Application Planning for a Siebel Upgrade."](#)
- 5 [Appendix A, "Siebel Marketing Upgrade Reference."](#)
- 6 [Appendix B, "Tables Modified or Seeded During a Siebel Upgrade."](#)

### Remote Upgrade Planning

See the following resources to deploy the upgrade to remote users on Siebel Mobile Web Clients:

- Use Siebel Packager to create language packs for remote installation. See *Going Live with Siebel Business Applications*.
- Use Siebel Anywhere to create installation kits for deployment. See *Siebel Anywhere Administration Guide*.
- Generate a Siebel Remote database template. See *Siebel Remote and Replication Manager Administration Guide*.
- Set up database extraction for Mobile Web Clients. See *Siebel Remote and Replication Manager Administration Guide*.

# Process of Upgrading a Siebel Development Environment

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** Windows and UNIX only. This topic does not apply to IBM z/OS.

This process is part of a road map. See [“Road Map for Performing a Siebel Database Upgrade” on page 37](#).

This topic lists the steps required to upgrade a Siebel development environment to the current release. Print this topic and use it as a checklist for doing the upgrade.

The topic is divided into sections, each containing a list of numbered steps. Complete each section in the order shown.

Steps that apply only to upgrades from Siebel 6.x are marked as follows: 6.x upgrades only.

**CAUTION:** Do not use this topic to perform a development environment upgrade on IBM z/OS. Instead, refer to *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

## Check SupportWeb

- 1 Check Siebel SupportWeb for recently published Technical Notes and Alerts regarding upgrade:
  - SupportWeb > Installation and Upgrade
  - SupportWeb > Technical Notes > Installation; Upgrade; Configuration-Dev Env; Configuration-General
  - SupportWeb > Siebel Alerts > Installation; Upgrade; Configuration Dev Env; Configuration-General
  - [“About Siebel Upgrade Planning Resources” on page 115](#)

## Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are located on Siebel SupportWeb under Product Documentation > Maintenance Release Guides.

**CAUTION:** Do not install a new Siebel database as part of upgrading the Siebel Enterprise.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using.

- 2 Upgrade the Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).  
For information upgrading these Siebel Enterprise components, see *Siebel Installation Guide* for the operating system you are using.
- 3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade. You only need to install the database server files for the database type that you are upgrading.

- 4 Install language packs for your currently deployed languages and any new languages.
- 5 If you have customized the configuration of Enterprise components, such as Siebel Servers, you must manually enter the customizations in the upgraded environment. See *Going Live with Siebel Business Applications*.

## Upgrade Third-Party Software

- 6 Upgrade third-party software as required due to dependencies on Oracle's Siebel software or other installed software. For example, you may need to upgrade the following software:
  - Actuate Server (Siebel Reports Server).
  - Operating system software. Some database upgrades require newer versions of AIX or Windows.

## Upgrade the RDBMS

- 7 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade:
  - For information on supported RDBMS systems, see *System Requirements and Supported Platforms* on Siebel SupportWeb.
  - For information on how upgrading your RDBMS affects the upgrade process, see ["About Upgrading Your RDBMS in the Siebel Environment"](#) on page 121.

## Pre-Upgrade Tasks for the Siebel Database

These steps apply to all database types.

- 8 Review guidelines for configuring the RDBMS. See the *Siebel Installation Guide* for the operating system you are using.
- 9 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 10 Stop the Siebel Servers and the Siebel Gateway Name Server.
- 11 Perform the tasks in [Chapter 7, "Basic Database Preparations for a Siebel Upgrade."](#)

## Pre-Upgrade Tasks for IBM DB2

- 12 Perform the tasks in [Chapter 8, "Preparing an IBM DB2 Database for a Siebel Upgrade."](#)
- 13 Execute runstats on the Siebel database. This will improve upgrade performance.

## Pre-Upgrade Tasks for Oracle

- 14 Perform the tasks in [Chapter 9, "Preparing an Oracle Database for a Siebel Upgrade."](#)

- 15 Run the Oracle Analyze command on the Siebel database. Highly fragmented indexes can cause the upgrade to fail.

### Pre-Upgrade Tasks for MS SQL Server

- 16 Perform the tasks in [Chapter 10, "Preparing an Microsoft SQL Server Database for a Siebel Upgrade."](#)
- 17 Run Microsoft SQL Server statistics. This will improve upgrade performance.

### Pre-Upgrade Tasks for Application Data

- 18 Perform the tasks in Technical Note 521 on Siebel SupportWeb. [Table 17](#) shows the applicability of the tasks in the Technical Note.

Table 17. Tasks in Technical Note 521

Item	Environment
Location and amendment of S_VIEW_WTMPL_IT.ITEM_NUM records	Development, production test, production
Update multi-value group (MVG) and association list applets	Development only
Update unnamed controls in applet web template items	Development only

- 19 Perform the tasks in the Technical Notes and Alerts on Siebel SupportWeb shown in [Table 18](#).

Table 18. Tasks in Technical Notes and Alerts

Title	Environment
Technical Note 481	Development only
Alert 1113	Development only
Technical Note 580	Development only

- 20 Perform the tasks in [Chapter 11, "Preparing Siebel Application Data for Upgrade."](#)

### Preparing Developers for the Upgrade

- 21 Back up the development database.
- 22 Verify that all developers have checked in their projects and that all projects are unlocked.

**23** Verify that all developers have disconnected from the database. The only open account must be the account that is performing the upgrade.

**24** Install the new Siebel Tools and license keys on development workstations.

**NOTE:** Keep at least one copy of the previous version of Siebel Tools. You will need it to perform repository operations before the repository merge.

Perform all remaining tasks with the new Siebel Tools, unless stated otherwise.

## Upgrade Siebel Database Schema (upgrep)

**25** [“Renaming the Siebel Repository” on page 203.](#)

**26** (Optional) [“Changing the Siebel Database Server Configuration Utilities Language” on page 204.](#)

**27** Run the Database Configuration Wizard:

- [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
- [“Running the Siebel Database Configuration Wizard Under Windows” on page 208.](#)
- [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209.](#)

Choose the following settings:

- **Upgrade Options:** Upgrade Siebel database Schema (upgrep)
- **Environment Type:** Development

**28** Edit generated SQL files as required by Siebel Technical Notes, Alerts, Release Notes or other publications. For a complete list, see the Installation and Upgrade Portal page on Siebel SupportWeb.

**29** Resume the upgrade: [“Starting the Siebel Upgrade Wizard” on page 211.](#)

**30** Review the upgrade logs and resolve errors:

- [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)
- [“Reviewing Siebel Upgrade Log Files for Errors” on page 221.](#)

**31** If the upgrade contains unacceptable errors, do the following:

- a Restore the backup of the database.
- b Correct the errors.
- c If errors occurred because you entered incorrect information in the Database Configuration Wizard, see [“Regenerating SQL Files for a Siebel Upgrade” on page 215.](#)
- d Rerun the Database Configuration Wizard.

**32 Multilingual deployments:** Perform the steps in Technical Note 447 on Oracle’s Siebel SupportWeb to import language-specific repository strings and seed data into the upgrade repositories.

**33** Back up the upgraded database.

## Prepare for Repository Merge

- 34 **6.x upgrades only:** [“Running the Siebel Repository Preparation Wizard” on page 245.](#)
- 35 **6.x upgrades only:** If the Repository Preparation Wizard log files contain unacceptable errors, do the following:
  - a Restore the backup of the upgraded database.
  - b Correct the errors.
  - c Rerun the Repository Preparation Wizard.
- 36 Set the Upgrade Ancestor property for copied objects. See [“Configuring Siebel Repository Objects to Inherit Upgrade Behavior” on page 227.](#)
- 37 See [“Configuring Siebel Tools for the Repository Merge” on page 228.](#)
- 38 (IBM DB2 and Microsoft SQL Only) Run database statistics on the Siebel database. If upgrading from Siebel 7.0.x or 7.5.x, run statistics specifically on S\_SYM\_STR and S\_SYM\_STR\_INTL.  
Running statistics on the Siebel database improves merge performance.

## Perform Repository Merge

Use the upgraded version of Siebel Tools to perform these tasks.

- 39 [“Performing a Siebel Repository Merge” on page 230.](#)
- 40 [“Reviewing the Siebel Repository Merge Log Files” on page 235.](#)
- 41 If the repository merge contains unacceptable errors, do the following:
  - a Restore the backup of the upgraded database.
  - b Correct the errors.
  - c Rerun the repository merge.
- 42 [“Generating Siebel EIM Temporary Columns” on page 237.](#)
- 43 Reset Siebel Tools configuration. See [“Configuring Siebel Tools for the Repository Merge” on page 228.](#)
- 44 Back up the Siebel database.

## Upgrade Custom Database Schema (upgphys)

- 45 Run the Database Configuration Wizard:
  - [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
  - [“Running the Siebel Database Configuration Wizard Under Windows” on page 208.](#)
  - [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209.](#)

Choose the following settings:

- **Upgrade Options:** Upgrade Custom Database Schema (upgphys)

- **Environment Type:** Development
- 46 Run the Upgrade Wizard: [“Starting the Siebel Upgrade Wizard” on page 211.](#)
- 47 Review the upgrade log files and resolve errors:
  - [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)
  - [“Reviewing Siebel Upgrade Log Files for Errors” on page 221.](#)
- 48 If the upgrade contains unacceptable errors, do the following:
  - a Restore the backup of the database you made after the repository merge.
  - b Correct the errors.
  - c Repeat the steps beginning after the backup.
- 49 [“Manually Archiving Siebel Upgrade Log Files” on page 226.](#)
- 50 Back up the upgraded database.

## Initialize and Extract Developers’ Local Databases

- 51 Install the new release of Siebel Tools on developers’ machines.
- 52 Initialize and extract the local database on development machines. For more information, see *Developing and Deploying Siebel Business Applications*.

## Migrate Created or Modified Applets and Views

**6.x upgrades only:** The steps in [“Migrate Created or Modified Applets and Views”](#) apply only to upgrades from Siebel 6.x.

- 53 (Optional) [“Identifying What Will Be Migrated in a Siebel 6.x Upgrade” on page 254.](#)
- 54 [“Exposing Hidden Properties of Siebel Applets and Views” on page 255.](#)
- 55 [“Running the Siebel Web Client Migration Wizard” on page 256.](#)
- 56 [“Reviewing Siebel Applet and View Migration” on page 259](#)
- 57 Remigrate applets and views as needed. See [“Remigrating Siebel Applets and Views” on page 260.](#)
- 58 [“Regenerating the Siebel Postmerge Utilities Log” on page 261.](#)

## Review the User Interface

- 59 [“Reviewing Siebel Repository Object Property Conflicts” on page 238](#)
- 60 Review the following topics on UI upgrade before proceeding:
  - [“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94](#)
  - [“About the Siebel Postmerge Utilities” on page 97](#)
  - For ICL upgrades: [“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

- 61 If you performed an ICL upgrade and chose Label on Top, revise the cascading style sheet. See [“Setting Label Alignment for Siebel Text Fields”](#) on page 242.
- 62 Create a new SRF file to assist in UI testing. For information on creating an SRF file, see *Developing and Deploying Siebel Business Applications*.
- 63 If you customized style sheet or Web template files in the previous release, reimplement those customizations in the new release as desired.  
Carefully review the UI in the new release before reimplementing customizations to these files.
- 64 Perform the tasks in [Chapter 16, “Reviewing the Siebel User Interface.”](#)
- 65 The postmerge utilities do not convert certain types of flow-based applets to grid-based applets. For example, they do not convert custom form applets to grid-based applets. See the “Editing Applet Layout” portion of *Configuring Siebel Business Applications* and convert remaining flow-based applets as desired.

## Postmerge Development Tasks

- 66 **6.x upgrades only:** Perform the tasks in [Chapter 17, “Postmerge Development Tasks for Siebel 6.x Upgrades.”](#)
- 67 Perform the tasks in [Chapter 18, “Siebel Postmerge Development Tasks.”](#)
- 68 Resolve any business component and join conflicts.
- 69 [“Deleting Unneeded Siebel Repository Files”](#) on page 245.

## Postupgrade Tasks for Database and File System

- 70 Perform the tasks in [Chapter 19, “Postupgrade Tasks for the Siebel Database and File System.”](#)
- 71 Reset upgrade-specific database and database server parameters back to their recommended settings for production. See *Siebel Installation Guide* for the operating system you are using for recommended parameter settings.
- 72 If you exported data from interface tables before the upgrade, review the database and import the data as desired. See *Siebel Enterprise Integration Manager Administration Guide*.
- 73 [“Upgrade Planning for Siebel RC2 or AES Encryption”](#) on page 132.
- 74 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide*.
- 75 (Oracle database 9i and later). Set Optimizer Mode to CBO. See [“Setting Oracle Database Configuration Parameters for a Siebel Upgrade”](#) on page 170.
- 76 Run database statistics.

**NOTE:** The upgrade is complete. The remaining topics deal with configuration and validation tasks.

## Postupgrade Tasks for Applications Configuration

- 77 [“Installing New Siebel License Keys During an Upgrade”](#) on page 217.

- 78 If applicable, review the results of the Person and Organization merge. Make configuration changes as required. To determine applicability or obtain more information, see [“About the Siebel Party Model” on page 124](#).
- 79 Perform the tasks in [Chapter 20, “Postupgrade Tasks for Siebel Applications.”](#)
- 80 Verify the function of interfaces in integrated applications.
- 81 Activate and deploy workflows.
- 82 If you have set up integrations for transferring data to or from third-party applications using Siebel EAI, verify the integrations are configured correctly. For information on EAI, see *Overview: Siebel eBusiness Application Integration Volume I*.
- 83 If you have used EIM to set up batch processing jobs, verify EIM is configured correctly. For information on EIM, see *Siebel Enterprise Integration Manager Administration Guide*.

## System Tests

- 84 Use available test data to perform unit testing. Validate the operation of the application in the following areas:
  - User interface
  - Data interfaces
  - Integrity of migrated data
  - Workflow function

## Prepare for Transition to Production Test Environment

- 85 Create a new SRF file. For information on creating an SRF file, see *Developing and Deploying Siebel Business Applications*.
- 86 If you revised the repository after running upgphys, you must regenerate the repository definition files. See [“Regenerating the Siebel Repository Definition Files” on page 240](#).

# Process of Upgrading a Siebel Production Test Environment

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only. This topic does not apply to IBM z/OS.

This process is part of a road map. See [“Road Map for Performing a Siebel Database Upgrade” on page 37](#).

This topic lists the steps required to upgrade a production test environment to the current release. Print this topic and use it as a checklist for doing the upgrade.

The topic is divided into sections, each containing a list of numbered steps. Complete each section in the order shown.

**CAUTION:** Do not use this topic to perform a production test environment upgrade on IBM z/OS. Instead, refer to *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

## Check SupportWeb

- 1 Check Siebel SupportWeb for recently published Technical Notes and Alerts regarding upgrade:
  - SupportWeb > Installation and Upgrade
  - SupportWeb > Technical Notes > Installation; Upgrade; Configuration-Dev Env; Configuration-General
  - SupportWeb > Siebel Alerts > Installation; Upgrade; Configuration Dev Env; Configuration-General
  - [“About Siebel Upgrade Planning Resources” on page 115](#)

## Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are located on Siebel SupportWeb under Product Documentation > Maintenance Release Guides.

**CAUTION:** Do not install a new Siebel database as part of upgrading the Siebel Enterprise.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using.

- 2 Upgrade the Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).  
For information upgrading these Siebel Enterprise components, see *Siebel Installation Guide* for the operating system you are using.
- 3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade. You only need to install the database server files for the database type that you are upgrading.
- 4 Install language packs for your currently deployed languages and any new languages.
- 5 If you have customized the configuration of Enterprise components, such as Siebel Servers, you must manually enter the customizations in the upgraded environment. See *Going Live with Siebel Business Applications*.

## Upgrade Third-Party Software

- 6 Upgrade third-party software as required due to dependencies on Oracle's Siebel software or other installed software. For example, you may need to upgrade the following software:
  - Actuate Server (Siebel Reports Server).
  - Operating system software. Some database upgrades require newer versions of AIX or Windows.

## Upgrade the RDBMS

- 7 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade:
  - For information on supported RDBMS systems, see *System Requirements and Supported Platforms* on Siebel SupportWeb.
  - For information on how upgrading your RDBMS affects the upgrade process, see ["About Upgrading Your RDBMS in the Siebel Environment"](#) on page 121.

## Pre-Upgrade Tasks for the Siebel Database

These steps apply to all database types.

- 8 Review guidelines for configuring the RDBMS. See the *Siebel Installation Guide* for the operating system you are using.
- 9 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 10 Perform the tasks in [Chapter 7, "Basic Database Preparations for a Siebel Upgrade."](#)

## Pre-Upgrade Tasks for IBM DB2

- 11 Perform the tasks in [Chapter 8, "Preparing an IBM DB2 Database for a Siebel Upgrade."](#)
- 12 Execute runstats on the Siebel database. This will improve upgrade performance.

## Pre-Upgrade Tasks for Oracle

- 13 Perform the tasks in [Chapter 9, "Preparing an Oracle Database for a Siebel Upgrade."](#)
- 14 Run the Oracle Analyze command on the Siebel database. Highly fragmented indexes can cause the upgrade to fail.

## Pre-Upgrade Tasks for Microsoft SQL Server

- 15 Perform the tasks in [Chapter 10, "Preparing an Microsoft SQL Server Database for a Siebel Upgrade."](#)
- 16 Run Microsoft SQL Server statistics. This will improve upgrade performance.

## Pre-Upgrade Tasks for Application Data

- 17** Perform the tasks in Technical Note 521 on Siebel SupportWeb. [Table 19](#) shows the applicability of the tasks in the Technical Note.

Table 19. Tasks in Technical Note 521

Item	Environment
Location and amendment of S_VIEW_WTMPL_IT.ITEM_NUM records	Development, production test, production
Update multi-value group (MVG) and association list applets	Development only
Update unnamed controls in applet web template items	Development only

- 18** Perform the tasks in the Technical Notes and Alerts on Siebel SupportWeb shown in [Table 20](#).

Table 20. Tasks in Technical Notes and Alerts

Title	Environment
Technical Note 481	Development only
Alert 1113	Development only
Technical Note 580	Development only

- 19** Perform the tasks in [Chapter 11, "Preparing Siebel Application Data for Upgrade."](#)

## Prepare the Siebel Database for Upgrade

- 20** Refer to Technical Note 586 on Siebel SupportWeb for instructions on how to set up the Siebel database and Siebel Servers in the production test environment.
- 21** Stop the Siebel Servers and the Siebel Gateway Name Server.
- 22** Close all database connections. The only database connection must be the account performing the upgrade.
- 23** Copy application files to the environment:
- a** Custom SRF file.
  - b** Reports files.
  - c** Custom Web templates and style sheets. See ["Copying UI Files to a New Siebel Environment"](#) on [page 243](#).

- 24 Copy the custrep.dat and schema.ddl files to the environment. See [“Moving the Siebel Repository Files” on page 214](#)

If you have made changes to the Siebel Repository since performing the development upgphys, you must regenerate the schema.ddl and custrep.dat files. See [“Regenerating the Siebel Repository Definition Files” on page 240](#).

- 25 Verify the production test database is either a copy of the current production database or has the same topology and amount of data.

This is important for effective upgrade tuning before performing the production upgrade.

- 26 Back up the production test environment database. (If you backed up the database as part of an RDBMS upgrade, ignore this step.)

To do upgrade tuning, you will restore this database and perform test-upgrades on it.

## Run the Upgrade Utilities in Prepare for Production Mode

- 27 (Optional) [“Changing the Siebel Database Server Configuration Utilities Language” on page 204](#).

- 28 Run the Database Configuration Wizard:

- [“Preparing to Run the Siebel Database Configuration Wizard” on page 205](#).
- [“Running the Siebel Database Configuration Wizard Under Windows” on page 208](#).
- [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209](#).

Choose the following settings:

- **Upgrade Options:** Prepare for Production Upgrade
- **Environment Type:** Production

- 29 Review the following topic and perform the procedure if applicable: [“Identifying and Dropping Obsolete Indexes for a Siebel Upgrade” on page 215](#).

## Upgrade the Siebel Database (upgrep)

- 30 Run the Database Configuration Wizard:

- [“Preparing to Run the Siebel Database Configuration Wizard” on page 205](#).
- [“Running the Siebel Database Configuration Wizard Under Windows” on page 208](#).
- [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209](#).

Choose the following settings:

- **Upgrade Options:** Upgrade Siebel Database Schema (upgrep)
- **Environment Type:** Production

- 31 Edit generated SQL files as required by Siebel Technical Notes, Alerts, Release Notes or other publications. For a complete list, see the Installation and Upgrade Portal page on Siebel SupportWeb.

- 32 Resume the upgrade: [“Starting the Siebel Upgrade Wizard” on page 211.](#)
- 33 Review the upgrade logs and resolve errors:
  - [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)
  - [“Reviewing Siebel Upgrade Log Files for Errors” on page 221.](#)
- 34 If the upgrade contains unacceptable errors, do the following:
  - a Restore the backup of the database.
  - b Correct the errors.
  - c If errors occurred because you entered incorrect information in the Database Configuration Wizard, see [“Regenerating SQL Files for a Siebel Upgrade” on page 215.](#)
  - d Rerun the Database Configuration Wizard.
- 35 **Multilingual deployments:** Perform the steps in Technical Note 447 on Oracle’s Siebel SupportWeb to import language-specific repository strings and seed data into the upgrade repositories.

## Upgrade the Siebel Database (upgphys)

- 36 Run the Database Configuration Wizard:
  - [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
  - [“Running the Siebel Database Configuration Wizard Under Windows” on page 208.](#)
  - [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209.](#)

Choose the following settings:

- **Upgrade Options:** Upgrade Custom Database Schema (upgphys)
  - **Environment Type:** Production
- 37 Run the Upgrade Wizard: [“Starting the Siebel Upgrade Wizard” on page 211.](#)
  - 38 Review the upgrade logs and resolve errors:
    - [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)
    - [“Reviewing Siebel Upgrade Log Files for Errors” on page 221.](#)
  - 39 If the upgrade contains unacceptable errors, do the following:
    - a Restore the backup of the database.
    - b Correct the errors.
    - c Rerun the Database Database Configuration Wizard.
  - 40 [“Manually Archiving Siebel Upgrade Log Files” on page 226.](#)
  - 41 [“Installing New Siebel License Keys During an Upgrade” on page 217.](#)
  - 42 Back up the upgraded database.

## Postupgrade Tasks for Database and File System

- 43 Perform the tasks in [Chapter 19, “Postupgrade Tasks for the Siebel Database and File System.”](#)
- 44 Reset upgrade-specific database and database server parameters back to their recommended settings for production. See *Siebel Installation Guide* for the operating system you are using for recommended parameter settings.
- 45 If you exported data from interface tables before the upgrade, review the database and import the data as desired.
- 46 [“Upgrade Planning for Siebel RC2 or AES Encryption” on page 132.](#)
- 47 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide*.
- 48 (Oracle database 9i and later). Set Optimizer Mode to CBO. See [“Setting Oracle Database Configuration Parameters for a Siebel Upgrade” on page 170.](#)
- 49 Run database statistics.

**NOTE:** The upgrade is complete. The remaining topics deal with configuration and validation tasks.

## Postupgrade Tasks for Applications Configuration

- 50 **6.x upgrades only:** If applicable, review the results of the Person and Organization merge. Make configuration changes as required. To determine applicability or obtain more information, see [“About the Siebel Party Model” on page 124.](#)
- 51 Perform the tasks in [Chapter 20, “Postupgrade Tasks for Siebel Applications.”](#)
- 52 Verify the function of interfaces in integrated applications.
- 53 Activate and deploy workflows.
- 54 If you have set up integrations for transferring data to or from third-party applications using Siebel EAI, verify the integrations are configured correctly. For information on EAI, see Overview: Siebel eBusiness Application Integration Volume I.
- 55 If you have used EIM to set up batch processing jobs, verify EIM is configured correctly. For information on EIM, see *Siebel Enterprise Integration Manager Administration Guide*.
- 56 If you customized style sheet or Web template files in the previous release, reimplement those customizations in the new release as desired.  
  
Carefully review the UI in the new release before reimplementing customizations to these files.

## System Tests

- 57 Use available test data to perform unit testing. Validate the operation of the application in the following areas:
  - User interface
  - Data interfaces
  - Integrity of migrated data

- Workflow function

## Process of Tuning Siebel Upgrade Performance

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only. This topic does not apply to IBM z/OS.

This process is optional.

This process is part of a road map. See [“Road Map for Performing a Siebel Database Upgrade” on page 37](#).

Use this process to run test upgrades in the production test environment so you can tune upgrade performance. Improving upgrade performance reduces downtime when you perform the production environment upgrade. The steps in this process cover standard performance tuning. To implement more advanced tuning, including high-availability tuning, contact Oracle’s Siebel Expert Services.

Perform this process in the production test environment. Do not perform this process in the production environment.

Review the following upgrade planning and performance tuning resources before performing this process:

- **Technical Note 616.** This Technical Note describes the strategies for minimizing production environment downtime during an upgrade. The steps that follow are intended primarily for use with the Baseline best practices described in Technical Note 616.
- [Chapter 21, “Tuning the Siebel Upgrade Files.”](#) This chapter provides detailed information on how to use Upgrade Tuner.

**UNIX users.** You must have a Siebel Server installed on a Windows host to run the Upgrade Tuner. To obtain a Windows version of Siebel Server, contact your account manager or Oracle.

**CAUTION:** Do not use this topic to tune upgrade performance on IBM z/OS. Instead, refer to *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

### Set Up the Target Database

- 1 Back up and remove the upgraded production test database.

This preserves seed data and metadata changes you have made that must be migrated to the production environment. This database is called the **production-test final database**.

- 2 In the production test environment, install a recent backup of your production database.

This database has not been upgraded and is called the **target database**. You will use it to perform test upgrades as part of tuning upgrade performance.

- 3 Define an ODBC connection to the target database.

- 4 Verify that the target database and RDBMS server are configured for optimum upgrade performance:
  - Chapter 8, "Preparing an IBM DB2 Database for a Siebel Upgrade."
  - Chapter 9, "Preparing an Oracle Database for a Siebel Upgrade."
  - Chapter 10, "Preparing an Microsoft SQL Server Database for a Siebel Upgrade."
- 5 "Preparing Siebel Tables and Views for Upgrade" on page 151.
- 6 Run statistics on the target database. This optimizes query performance.
- 7 **UNIX users.** "Securing AIX Memory Allocation Segment Space for the Siebel Database" on page 154.
- 8 Perform the tasks in Chapter 11, "Preparing Siebel Application Data for Upgrade."

## Upgrading the Target Database (upgrep)

- 9 (Optional) "Changing the Siebel Database Server Configuration Utilities Language" on page 204.
- 10 In the production test environment, run the Database Configuration Wizard:
  - "Preparing to Run the Siebel Database Configuration Wizard" on page 205.
  - "Running the Siebel Database Configuration Wizard Under Windows" on page 208.
  - "Running the Siebel Database Configuration Wizard Under UNIX" on page 209.Choose the following settings:
  - **Upgrade Options:** Upgrade Siebel Database Schema (upgrep)
  - **Environment Type:** Production
- 11 In the production test environment, run Siebel Upgrade Wizard.  
See "Starting the Siebel Upgrade Wizard" on page 211.  
Note the time required to upgrade the database.
- 12 Review the upgrade log files for errors:
  - "Summarizing Siebel Log Files Using the Logparse Utility" on page 219.
  - "Reviewing Siebel Upgrade Log Files for Errors" on page 221.
- 13 If the upgrade contains errors that prevented completion or adversely affected performance, correct the errors and rerun the upgrade.
- 14 "Manually Archiving Siebel Upgrade Log Files" on page 226.

## Tuning the Upgrade Files

- 15 Evaluate upgrade performance, particularly the time required to complete the upgrade.
- 16 Do one of the following:

- If the time required to complete the upgrade is acceptable, no further tuning is needed. Perform the steps in [“Restore the Production-Test Final Database”](#).
  - If the time required to complete the upgrade is too long, perform the remaining steps in this topic to continue tuning upgrade performance.
  - If the time required to complete the upgrade is too long and you cannot tune further, contact Oracle’s Siebel Expert Services to apply advanced tuning.
- 17** Carefully review target database and RDBMS server configuration. Adjust as needed to further improve upgrade performance.
- 18** Run Upgrade Tuner to tune the upgrade files:
- **UNIX users.** [“Transferring UNIX Files for Use by Siebel Upgrade Tuner”](#) on page 351.
  - [“Starting and Stopping Siebel Upgrade Tuner”](#) on page 343.
  - [“Managing Parallel Threads Using Siebel Upgrade Tuner”](#) on page 345.
  - [“Managing Zero-Row SQL Commands Using Siebel Upgrade Tuner”](#) on page 348.
- To roll back changes you have made in previous Upgrade Tuner sessions, see [“Rolling Back Siebel Upgrade Tuner Changes”](#) on page 354.

## Restore the Target Database

Perform these steps if you have made changes to the upgrade environment or to the upgrade files and want to run the upgrade again to verify performance improvement.

- 19** In the production test environment, restore the target database from backup.
- This returns the target database to its non-upgraded state so that you can perform another test upgrade.
- 20** Perform another test upgrade and evaluate upgrade performance.
- 21** Repeat the tuning process and perform test-upgrades until upgrade performance is acceptable.

## Restore the Production-Test Final Database

Perform these steps only if you have completed tuning upgrade performance.

- 22** In the production test environment, delete and remove the target database.
- 23** In the production test environment, restore the production-test final database from backup.

# Process of Upgrading a Siebel Production Environment

**Upgrades from:** All Siebel releases.

**Environments:** Production environment.

**Platforms:** Windows and UNIX only. This topic does not apply to IBM z/OS.

To perform this process, you must be able to execute ODBC commands on the production database from within the production test environment. For more information, see [“About the Siebel Database Upgrade Process” on page 68](#). If you cannot do this, contact Siebel Technical Support.

This process is part of a road map. See [“Road Map for Performing a Siebel Database Upgrade” on page 37](#).

This topic lists the tasks required to transition your production test environment to production. Print this topic and use it as a checklist for doing the upgrade. It is divided into sections, each containing numbered steps. Complete the steps in the order shown.

**CAUTION:** Do not use this topic to perform a production upgrade on IBM z/OS. Instead, see *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

## Check SupportWeb

- 1 Check Siebel SupportWeb for recently published Technical Notes and Alerts regarding upgrades:
  - SupportWeb > Installation and Upgrade
  - SupportWeb > Technical Notes > Installation; Upgrade; Configuration-Dev Env; Configuration-General
  - SupportWeb > Siebel Alerts > Installation; Upgrade; Configuration Dev Env; Configuration-General
  - [“About Siebel Upgrade Planning Resources” on page 115](#)

## Upgrade the Servers

Verify that you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are located on Siebel SupportWeb under Product Documentation > Maintenance Release Guides.

**CAUTION:** Do not install a new Siebel database as part of upgrading the Siebel Enterprise.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using.

- 2 Upgrade the Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).  
For information upgrading these Siebel Enterprise components, see *Siebel Installation Guide* for the operating system you are using.
- 3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade. You only need to install the database server files for the database type that you are upgrading.
- 4 Install the language packs for your currently deployed languages and any new languages.
- 5 If you have customized the configuration of Enterprise components, such as Siebel Servers, you must manually enter the customizations in the upgraded environment. See *Going Live with Siebel Business Applications*.

## Upgrade Third-Party Software

- 6 Upgrade third-party software as required due to dependencies on Oracle's Siebel software or other installed software. For example, you may need to upgrade the following software:
  - Actuate Server (Siebel Reports Server).
  - Operating system software. Some database upgrades require newer versions of AIX or Windows.

## Upgrade the RDBMS

- 7 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade:
  - For information on supported RDBMS systems, see *System Requirements and Supported Platforms* on Siebel SupportWeb.
  - For information on how upgrading your RDBMS affects the upgrade process, see ["About Upgrading Your RDBMS in the Siebel Environment"](#) on page 121.

## Pre-Upgrade Tasks for the Siebel Database

These steps apply to all database types.

- 8 Review guidelines for configuring the RDBMS. See the *Siebel Installation Guide* for the operating system you are using.
- 9 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 10 Stop the Siebel Servers and the Siebel Gateway Name Server.
- 11 Close all database connections. The only open connection must be the account performing the upgrade.
- 12 Perform the tasks in [Chapter 7, "Basic Database Preparations for a Siebel Upgrade."](#)

## Pre-Upgrade Tasks for IBM DB2

- 13 Perform the tasks in [Chapter 8, "Preparing an IBM DB2 Database for a Siebel Upgrade."](#)
- 14 Execute runstats on the Siebel database. This will improve upgrade performance.

## Pre-Upgrade Tasks for Oracle

- 15 Perform the tasks in [Chapter 9, "Preparing an Oracle Database for a Siebel Upgrade."](#)
- 16 Run the Oracle Analyze command on the Siebel database. Highly fragmented indexes can cause the upgrade to fail.

## Pre-Upgrade Tasks for Microsoft SQL Server

- 17 Perform the tasks in [Chapter 10, “Preparing an Microsoft SQL Server Database for a Siebel Upgrade.”](#)
- 18 Run Microsoft SQL Server statistics. This will improve upgrade performance.

## Pre-Upgrade Tasks for Application Data

- 19 Perform the tasks in Technical Note 521 on Siebel SupportWeb. [Table 21](#) shows the applicability of the tasks in the Technical Note.

Table 21. Tasks in Technical Note 521

Item	Environment
Location and amendment of S_VIEW_WTMPL_IT.ITEM_NUM records	Development, production test, production
Update multi-value group (MVG) and association list applets	Development only
Update unnamed controls in applet web template items	Development only

- 20 Perform the tasks in the Technical Notes and Alerts on Siebel SupportWeb shown in [Table 22](#).

Table 22. Tasks in Technical Notes and Alerts

Title	Environment
Technical Note 481	Development only
Alert 1113	Development only
Technical Note 580	Development only

- 21 Perform the tasks in [Chapter 11, “Preparing Siebel Application Data for Upgrade.”](#)

## Pre-Upgrade Tasks for the UI

- 22 Copy application files to the environment:
  - a Custom SRF file.
  - b Reports files.
  - c Custom Web templates and style sheets. See [“Copying UI Files to a New Siebel Environment” on page 243](#).

## Upgrade the Siebel Database (upgrep)

You do not run the upgrep and upgphys portions of the upgrade in the production environment. Instead, the numbered steps in this topic implement the following process:

- Run the Database Configuration Wizard in the production test environment.
- In the Utilities, enter the information for the production environment instead of the production test environment. For example, you enter the ODBC connection for the production environment.  
  
This information configures the driver file to run against the production database rather than the production test database. It also configures the driver file to use the upgrade SQL files you generated for the production test upgrade.
- Run the Upgrade Wizard. The Upgrade Wizard uses the SQL files in the production test environment to upgrade the database in the production environment.

This approach has several advantages:

- You do not have to generate upgrade SQL files in the production environment, and then manually transfer the customizations to them from the production test environment.
- You do not lose any changes to the SQL files that were made by Siebel Upgrade Tuner in the production test environment.
- You do not have to run the Database Configuration Wizard in Prepare for Production mode again.
- With some exceptions, you do not have to perform database-related configuration tasks required by Release Notes or Alerts again.

If your network configuration prevents creating an ODBC connection to your production database from inside your production test environment, contact Siebel Technical Support for assistance in completing the production upgrade.

- 23 Verify you have a current backup of the production environment database.
- 24 On the Siebel Server you used to upgrade the production test environment, create an ODBC to connect to the production environment database.
- 25 Navigate to *DBSRVR\_ROOT\common* (UNIX: *DBSRVR\_ROOT/common*) and verify that the file *sqlgen.usg* exists.

This file contains a record of when the SQL generator was run. When you run the Database Configuration Wizard, if this file exists, no SQL commands are generated.

**CAUTION:** If this file does not exist, do not run the Database Configuration Wizard. It will overwrite the SQL files used to upgrade your production test database. Contact Siebel Technical Support for guidance on proceeding with the upgrade.

You do not have to run the Database Configuration Wizard in Prepare for Production Mode.

- 26 (Optional) [“Changing the Siebel Database Server Configuration Utilities Language” on page 204.](#)
- 27 Run the Database Configuration Wizard:
  - [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
  - [“Running the Siebel Database Configuration Wizard Under Windows” on page 208.](#)
  - [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209.](#)



## Upgrade Custom Database Schema (upgphys)

You upgrade the production database by using the SQL scripts in the production test environment. In the production test environment, you run the Database Configuration Wizard but enter environment information for the production database. Then you start the Siebel Upgrade Wizard in the production test environment. The Siebel Upgrade Wizard uses the production database environment information and the SQL in the production test environment, including any changes you have made, to upgrade the production database.

- 34 Verify you have a current backup of the production environment database.
- 35 On the Siebel Server you used to upgrade the production test environment, verify you have created an ODBC to connect to the production environment database.
- 36 Run the Database Configuration Wizard:
  - [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
  - [“Running the Siebel Database Configuration Wizard Under Windows” on page 208.](#)
  - [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209.](#)Choose the following settings:
  - **Upgrade Options:** Upgrade Custom Database Schema (upgphys)
  - **Environment Type:** Production
- 37 Run the Upgrade Wizard: [“Starting the Siebel Upgrade Wizard” on page 211.](#)
- 38 Review the upgrade logs and resolve errors:
  - [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)
  - [“Reviewing Siebel Upgrade Log Files for Errors” on page 221.](#)
- 39 If the upgrade contains unacceptable errors, do the following:
  - a Restore the backup of the database.
  - b Correct the errors.
  - c Restart the Siebel Upgrade Wizard.
- 40 [“Manually Archiving Siebel Upgrade Log Files” on page 226.](#)
- 41 [“Installing New Siebel License Keys During an Upgrade” on page 217.](#)
- 42 Use the Siebel Application Deployment Manager to migrate administrative data such as LOVs and responsibilities from production test to production. See *Going Live with Siebel Business Applications*.
- 43 Back up the upgraded production environment database.

## Postupgrade Tasks for Database and File System

- 44 Perform the tasks in [Chapter 19, “Postupgrade Tasks for the Siebel Database and File System.”](#)

- 45 Reset upgrade-specific database and database server parameters back to their recommended settings for production. See *Siebel Installation Guide* for the operating system you are using for recommended parameter settings.
- 46 If you exported data from interface tables before the upgrade, review the database and import the data as desired.
- 47 [“Upgrade Planning for Siebel RC2 or AES Encryption” on page 132.](#)
- 48 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide*.
- 49 (Oracle database 9i and later). Set Optimizer Mode to CBO. See [“Setting Oracle Database Configuration Parameters for a Siebel Upgrade” on page 170.](#)
- 50 Run database statistics.

**NOTE:** The upgrade is complete. The remaining topics deal with configuration and validation tasks.

## Postupgrade Tasks for Applications Configuration

- 51 If applicable, review the results of the Person and Organization merge. Make configuration changes as required. To determine applicability or obtain more information, see [“About the Siebel Party Model” on page 124.](#)
- 52 Perform the tasks in [Chapter 20, “Postupgrade Tasks for Siebel Applications.”](#)
- 53 Verify the function of interfaces in integrated applications.
- 54 Activate and deploy workflows.
- 55 If you have set up integrations for transferring data to or from third-party applications using Siebel EAI, verify the integrations are configured correctly. For information on EAI, see *Overview: Siebel eBusiness Application Integration Volume I*.
- 56 If you have used EIM to set up batch processing jobs, verify EIM is configured correctly. For information on EIM, see *Siebel Enterprise Integration Manager Administration Guide*.
- 57 If you customized the style sheet or Web template files in the previous release, reimplement those customizations in the new release as desired.

## System Tests

- 58 Use available test data to perform unit testing. Validate the operation of the application in the following areas:
  - User interface
  - Data interfaces
  - Integrity of migrated data
  - Workflow function



# 4

## How the Siebel Database Upgrade Works

This chapter contains the following topics:

- [“About Supported Siebel Upgrade Paths” on page 65](#)
- [“About Siebel Upgrade Environments” on page 66](#)
- [“About the Siebel Database Upgrade Process” on page 68](#)
- [“About the Siebel Database Server Configuration Utilities” on page 72](#)
- [“About the Siebel Upgrade Wizard and Driver Files” on page 77](#)
- [“About the Siebel Database Upgrade Log Files” on page 80](#)
- [“ForExample of a Siebel Development Upgrade Flow” on page 82](#)
- [“About the Siebel Repository Merge” on page 89](#)
- [“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94](#)
- [“About the Siebel Postmerge Utilities” on page 97](#)
- [“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)
- [“About the Siebel Database Server” on page 109](#)
- [“About Tuning Siebel Production Upgrade Files” on page 109](#)

### Related Topics

[Chapter 2, “About Siebel Database Upgrade Topics”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## About Supported Siebel Upgrade Paths

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

Supported upgrade paths are described in *System Requirements and Supported Platforms* on Siebel SupportWeb.

The *Siebel Database Upgrade Guide* does not cover the following specific upgrade paths or infrastructure changes. Contact Siebel Technical Support or Siebel Expert Services for help with these tasks:

## How the Siebel Database Upgrade Works ■

- Direct upgrades from Siebel 6.x to releases after Siebel 7.7 are not supported, except as noted below. You must first upgrade from Siebel 6.x to Siebel 7.7.

However, direct upgrades from Siebel 6.2.1 of Siebel Financial Services to Siebel 7.8 or later of Siebel Industry Applications on IBM z/OS platforms are supported. To perform this upgrade, see *Siebel Database Upgrade Guide for DB2 UDB for z/OS*.

To upgrade from Siebel 6.x to Siebel 7.7, see the latest Siebel 7.7 version of the *Siebel Database Upgrade Guide* on Siebel SupportWeb.

- Changing database platform type during an upgrade, for example changing from Oracle 8i to IBM DB2.
- Changing operating system type during an upgrade, for example changing from Windows to UNIX.
- Migrating to Unicode.
- Migrating from Oracle's Siebel Industry Solutions applications or Siebel Financial Services to Siebel Business Applications.
- Upgrades from Siebel 6.x involving Siebel Connector for SAP, R/3.

If your enterprise uses SAP and you have implemented the Siebel Enterprise Application Integration (EAI) product, see *Siebel Connector for SAP R/3* for upgrade instructions.

- Upgrading from one base language to another. To achieve similar results, upgrade your existing base language and install the language pack for the desired language.

### Related Topics

["About Siebel Upgrade Environments" on page 66](#)

## About Siebel Upgrade Environments

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

The *Siebel Database Upgrade Guide* describes how to upgrade three environments:

- Development environment
- Production test environment
- Production environment

### Development Environment

A development environment contains the following Oracle modules:

- Siebel Server
- Web server and Siebel Web Server Extension
- Siebel Gateway Name Server

- Siebel Database Server files installed on a Siebel Server
- RDBMS server and Siebel database
- Siebel Tools installed on workstations running a supported Windows environment. This includes the local database running on developers' Mobile Web Clients.
- Siebel applications and test data required to verify the basic function of newly compiled SRF files.

The development environment is where developers use Siebel Tools to customize Siebel applications. The development environment upgrade merges these customizations with the new release. The merged repository and schema definitions become inputs to the production test environment upgrade and production upgrade.

### Production Test Environment

The production test environment includes the following Oracle modules:

- Siebel Enterprise, including at least one Siebel Server and an RDBMS server and Siebel database
- Web server with a Siebel Web Server Extension installed
- Siebel Gateway Name Server
- Siebel Database Server files installed on a Siebel Server
- All the Siebel applications currently installed in your production environment
- A copy of the Siebel database installed in your production environment

You perform the following processes in the production test environment:

- Test the upgraded release to validate its function and performance before deploying it to users.
- Tune the upgrade process to minimize the time required to perform your production upgrade.

Oracle provides an upgrade tuning application that analyzes how the upgrade scripts interacted with the production test environment database. The Upgrade Tuner enables you to tune how the scripts will execute against the Siebel database in your production environment. Tuning the scripts can significantly reduce the time required to complete the production upgrade. For this reason, the production test environment database must contain the same data volume and topography as your production database.

### Production Environment

The production environment is your live business environment, where your internal and external users interact with applications and generate actual business data. The production environment includes all your Siebel Enterprises worldwide.

The upgrade process assumes all production environment databases are completely separate from the development environment and production test environment databases.

Oracle provides these tools to help you transition from production test to production:

- **Siebel Application Deployment Manager (ADM)**. This module migrates administrative data such as lists of values (LOVs) from the production test environment to the production environment. For details, see *Going Live with Siebel Business Applications*.

- **Siebel Packager.** This application creates installation packages for use by Siebel Anywhere. For details, see *Going Live with Siebel Business Applications*.
- **Siebel Anywhere.** This application builds distribution kits for remote users. For details, see *Siebel Anywhere Administration Guide*.

### Mapping Your Environments

You may have more or fewer environments than those described above. [Table 23](#) gives recommendations for mapping your environments to the ones used in the *Siebel Database Upgrade Guide*.

Table 23. Mapping Your Environments to Upgrade Processes

Environment Description	Recommended Upgrade
<p>The environment has the following characteristics:</p> <ul style="list-style-type: none"> <li>■ It is used primarily for development with Siebel Tools.</li> <li>■ The Siebel database is a subset of your production database.</li> <li>■ The environment is not used for tech support or training. Developers are usually installed as Mobile Web Clients.</li> </ul>	Development environment upgrade.
<p>The environment has the following characteristics:</p> <ul style="list-style-type: none"> <li>■ It is intended for testing customizations before deploying them.</li> <li>■ It is where you tune your upgrade SQL files to minimize production upgrade time.</li> <li>■ There may be multiple upstream environments in addition to the production test environment. For example, these could include environments used by a product management group, Technical Support, and Quality Assurance.</li> </ul>	Production test environment upgrade.
<p>The environment is used for live business transactions by both local and remote users.</p>	Production environment upgrade.

#### Related Topics

[“About Supported Siebel Upgrade Paths” on page 65](#)

## About the Siebel Database Upgrade Process

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

Upgrading to a new release involves two aspects:

- The order in which to upgrade your environments
- The flow of the upgrade process within each environment

## Environment Upgrade Order

If you have a development environment, you must upgrade it first, before upgrading other environments. This merges your customizations with the new release. A merged repository file and database schema file are created and become inputs to the production test environment upgrade and production upgrade.

If you do not have a development environment or have not customized your repository, no repository merge is required. You can use the repository and schema definition files included in the new release to upgrade your production test environment and production environment.

## Flow of the Upgrade Within an Environment

The basic flow of the upgrade process is shown in [Figure 1](#). This flow applies to development environment and production test environment upgrades. You also use this flow to upgrade your production environment, with the addition of several deployment steps.

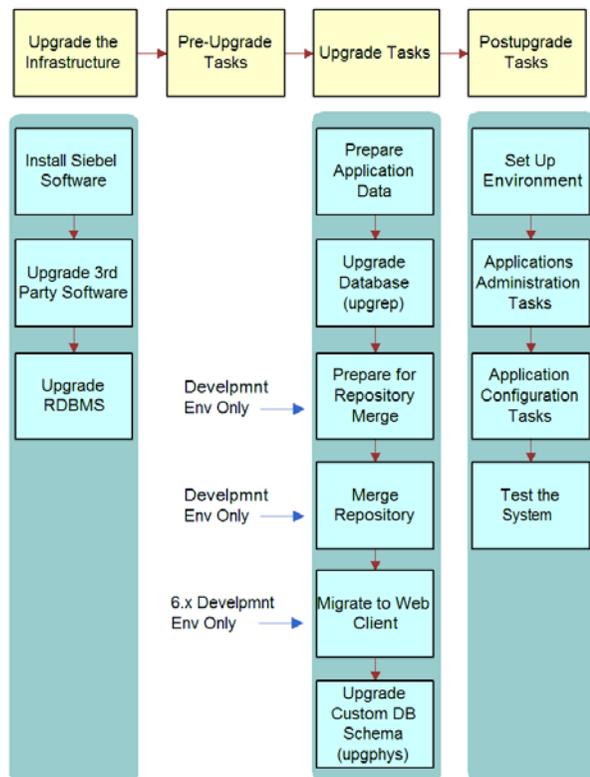


Figure 1. Flow of the Upgrade Process

### Upgrade the Infrastructure

The first phase is to upgrade your hardware and software to meet system and implementation requirements, which includes upgrading the Siebel Enterprise to the new release. This action upgrades the Siebel Servers and provides the programs, scripts, input files, and other files required to merge the repository and upgrade the Siebel database. For information on how to upgrade the infrastructure, see *Siebel Installation Guide* for the operating system you are using.

### Perform Pre-Upgrade Tasks

This phase prepares the Siebel database for upgrade and includes such tasks as closing the database connections, clearing pending workflow tasks, and disabling customized triggers.

### Perform Upgrade Tasks (Development Environment)

This phase merges your customizations into the new release. This phase also upgrades the development environment database and includes these tasks:

- **Prepare application data.** These tasks prepare test data for migration.
- **Upgrade database (upgrep).** You run the Database Configuration Wizard in upgrep mode. They perform a basic upgrade of the Siebel database schema and load repositories to prepare for the repository merge.
- **Prepare for repository merge.** If you are upgrading from Siebel 6.x, you run the Repository Preparation Wizard to migrate strings, merge field labels, and merge existing Web templates.
- **Merge repository.** You use Siebel Tools to merge your existing repository with the repository in the new release. Postmerge utilities upgrade form applets and verify that applets and views are configured correctly.
- **Migrate to Web Client.** You run the Web Client Migration Wizard to convert customizations in the Windows-based Siebel 6.x interface to the browser-based interface in Siebel 7.x.
- **Upgrade database (upgphys).** You run the Database Server Configuration Utilities in upgphys mode. They further upgrade the Siebel database with changes resulting from the repository merge and complete the database upgrade.

The Database Server Configuration Utilities also generate the customer repository definition file and logical schema definition file that are used as input to the production test environment and production upgrades.

**NOTE:** During the synchronization process, custom columns in the Siebel Schema that are not in the Siebel Repository are not deleted but custom indexes in the Siebel Schema that are not in the Siebel Repository are deleted.

### Perform Upgrade Tasks (Production Test Environment)

This phase upgrades a production test environment Siebel database to the new release and includes the following tasks:

- **Prepare application data.** These tasks are about preparing application data for migration.

- **Upgrade database (upgrep).** You run the Database Server Configuration Utilities in upgrep mode. They perform a basic upgrade of the Siebel database schema:
  - You run the utility in Prepare for Production mode before running it in upgrep mode. The Prepare for Production mode reviews the upgraded development environment database schema and creates input files. The upgrep mode uses these files to make schema changes to the Siebel database.
  - The upgrep mode imports the repository and schema definition files from the development environment upgrade. It uses these files to upgrade the Siebel database.
  -
- **Upgrade database (upgphys).** You run the Database Server Configuration Utilities in upgphys mode. This utility completes the upgrade and performs miscellaneous administrative tasks.

### Perform Upgrade Tasks (Production Environment)

This phase upgrades a production environment Siebel database to the new release and includes the following tasks:

- 
- **Prepare application data.** These tasks are about preparing application data in the production database for migration.
- **Upgrade database (upgrep).** In the production test environment, you start the Database Server Configuration Utilities in upgrep mode, and enter configuration information for the production environment. This includes an ODBC definition for connecting to the production Siebel database.

This step causes the wizard to use the SQL for upgrading the production test database to upgrade the production database.

The SQL generated for the production test database upgrade is preserved, and no new SQL is generated. This SQL has been tuned using the Upgrade Tuner and has been revised by the Prepare for Production mode. You do not have to repeat these two steps before upgrading the production database.

**Upgrade database (upgphys).** You run the Database Server Configuration Utilities in upgphys mode. You do this in the production test environment in the same fashion as running the upgrep. This utility completes the upgrade and performs miscellaneous administrative tasks.

### Perform Postupgrade Tasks

This phase is where you set up the environment, configure applications, and test the system as follows:

- **Set up the environment.** These tasks set up the postupgrade environment, which includes extracting the developers' databases and running database statistics.
- **Application administration.** These tasks set up the applications and include such tasks as setting up user access and visibility of views and screens.

- **Application configuration.** These tasks prepare applications for testing, including data migration for specific applications.

**Test the system.** These tasks test the system. For development environment upgrades, you perform basic unit tests to verify the operation of the application. For production test environment upgrades, you perform a full suite of regression and stress tests to verify the system is ready for production.

### Related Topics

[“About the Siebel Database Server Configuration Utilities” on page 72](#)

[“About the Siebel Upgrade Wizard and Driver Files” on page 77](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Database Server” on page 109](#)

# About the Siebel Database Server Configuration Utilities

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

The Database Server Configuration Utilities are part of the Siebel Configuration Wizard. The Database Server Configuration Utilities interactively gather the information required to perform the following operations:

- **Install the Siebel database.** These wizards set up the Siebel database in the RDBMS as part of a first-time installation of Siebel applications. They can also add a language to the Siebel Database Server installation.
- **Upgrade the Siebel database.** These wizards upgrade the Siebel Database to a new release in a development environment, production test environment, or production environment.
- **Import or export a Siebel repository.** These wizards move entire repositories between database environments with the same schema definition. A wizard is also provided that adds installed languages to a Tools repository.
- **Migrate a Siebel repository.** This wizard migrates repositories between databases and synchronizes the target database schema with the migrated repository schema definition.
- **Run database utilities.** This group of wizards perform the following functions:
  - Synchronize a database schema with the Siebel Repository schema definition.
  - Add a license key to a Siebel database (MSSQL only).
  - Convert existing time-keeping management to UTC.
  - Convert existing Lists of Values (LOVs) to Multilingual Lists of Values (MLOVs).

## Upgrade the Siebel Database

When you run the Database Configuration Wizard > Upgrade the Database, the wizard prompts you for the upgrade environment (development or production) and the upgrade phase (upgrep or upgphys). The Wizard then prompts you for the information it needs to perform the upgrade.

After collecting and confirming the information, the wizard creates an upgrade configuration file and calls a driver that uses the file to set up the SQL scripts required to upgrade your database.

After you run the Database Configuration Wizard, you run the Upgrade Wizard. The Upgrade Wizard opens a state file containing the steps for the upgrade. These steps execute the SQL scripts created after you ran the Database Configuration Wizard.

To upgrade a development environment, production test environment, and production environment, you must run the Database Configuration Wizard (and Upgrade Wizard) several times, as shown in [Table 24](#).

Table 24. Database Server Configuration Utilities Modes

Upgrade Step	Select This Environment Type	Select This Upgrade Option
Development env. upgrep	Development	upgrep
Development env. upgphys	Development	upgphys
Prod. test env. prepare for production	Production	Prepare for Production
Prod. test env. upgrep	Production	upgrep
Prod. test env. upgphys	Production	upgphys
Production env. upgrep	Production	upgrep
Production env. upgphys	Production	upgphys

Figure 2 shows how the Database Configuration Wizard (and Upgrade Wizard) work together with the Siebel Tools repository merge to upgrade your environments.

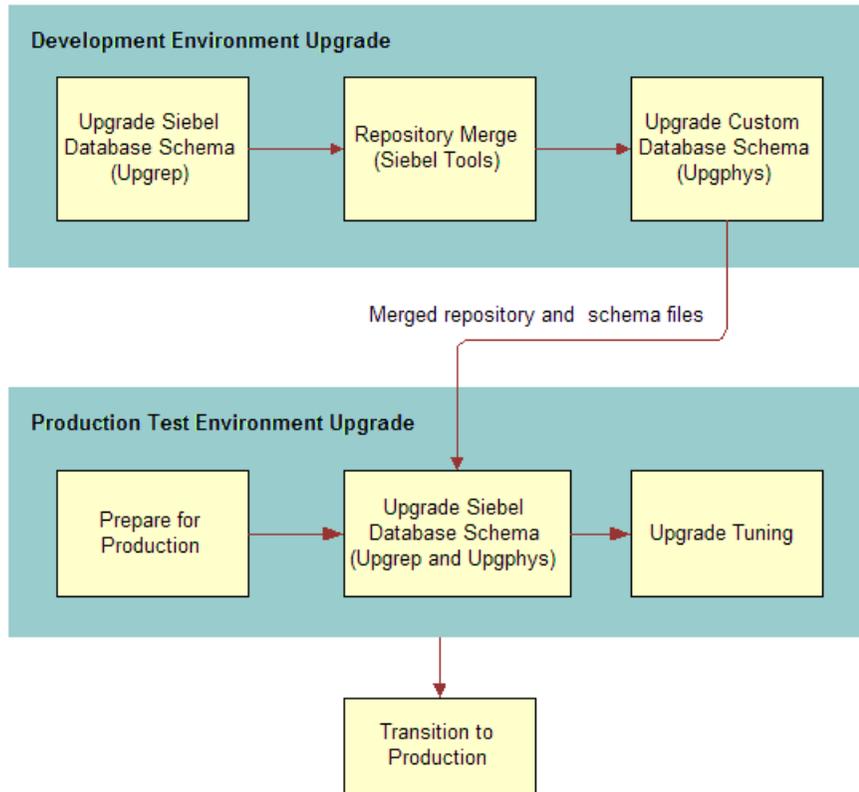


Figure 2. Upgrade Flow

### Development Environment Upgrades

For a development environment upgrade, you run the utility twice, once in each of the following modes:

- **upgrep.** This mode makes the following changes:
  - Drops interface tables and database triggers
  - Populates columns that must change from NULL to NOT NULL
  - Creates new tables. Merges existing tables.
  - Prepares for index creation. Verifies that there are no unique key violations.
  - Creates indexes
  - Imports seed data

- Imports the version x.x Prior Customer Repository, New Siebel Repository, and New Customer Repository
- Makes modifications to repository objects to prepare for the repository merge
- Updates primary children foreign key references
- Performs miscellaneous file actions
- **upgphys.** This mode makes the following changes:
  - Synchronizes the Siebel database schema to the logical schema definition in the merged repository
  - Deduplicates intersection tables
  - Exports repository object definitions to a file, custrep.dat
  - Exports the logical schema definition to a file, schema.ddlThese two files are used as input to the production test environment and production environment upgrades.
- Renames the New Customer Repository to Siebel Repository
- Updates the schema version in S\_APP\_VER

## Production Test Environment Upgrades

For a production test environment upgrade, you run the utility three times:

- **Prepare for production.** This mode does the following in the production test database:
  - Examines the development environment database and generates SQL that deduplicates intersection tables and sets up support for database aggregation.In the production test environment, you must define an ODBC connection to the development environment database before performing this upgrade step.
- Compares the repository schema and the physical database schema. Generates a file that lists indexes present in the physical schema that are not present in the repository schema. You can decide whether to drop these indexes.
- **upgrep.** This mode makes the following changes:
  - Drops interface tables and database triggers
  - Populates columns that must change from NULL to NOT NULL
  - Uses the custrep.dat and schema.ddl files from the development environment upgrade to create new tables and merge existing tables
  - Prepares for index creation. Verifies that there are no unique key violations.
  - Creates indexes
  - Imports seed data
  - Updates primary children foreign key references
  - Performs miscellaneous file actions

## How the Siebel Database Upgrade Works ■

- **upgphys.** This mode makes several administrative changes to table data, including updating the schema version in S\_APP\_VER. It does not make schema changes.
- **Upgrade Tuning.** These steps are optional. You can use the Logparse utility to tune the SQL generated for the upgrade. Logparse identifies SQL commands that are not used, or that can be run in parallel. This reduces the time required to perform the production database upgrade.

### Production Environment Upgrades

After you have completed testing applications and have tuned the upgrade SQL commands, you perform the production upgrade. The production upgrade uses the SQL commands generated in the production test environment. In the production test environment, you perform the following steps to upgrade your production environment Siebel database:

- Define an ODBC connection to the production environment Siebel database
- **Prepare for Production.** This step is not required. You ran it as part of the production test environment upgrade. The required changes have already been made to the upgrade SQL commands.
- **upgrep.** Run the Database Server Configuration Utilities in the production test environment. Enter information for the production environment (not the production test environment), including the new ODBC definition.

The Database Server Configuration Utilities update the upgrade configuration file with production environment information. A lock file that was set when you ran the utility previously, prevents new SQL from being generated. This preserves the SQL you have revised and tuned.

When you run the Upgrade Wizard, it reads the production environment information from the configuration file and uses the production test environment SQL commands to upgrade the production environment Siebel database.

The upgrep step makes the same changes as when it ran in the production test environment.

- **upgphys.** Run the Database Server Configuration Utilities in the production test environment. Enter information for the production environment (not the production test environment), including the new ODBC definition.

The Upgrade Wizard uses the upgrade SQL commands in the production test environment to upgrade the production environment Siebel Database.

The upgphys step makes the same changes as when it ran in the production test environment.

### How the Upgrade Configuration File and SQL Files Are Created

When you run the Database Server Configuration Utilities, they do the following:

- Collect configuration information
- Create a master upgrade configuration file (UCF). This file maps the information you entered in the Database Server Configuration Utilities to environment variables.
- Forward the information to an SQL generator that creates or populates SQL files with the required commands. The SQL generator extracts these commands from an intermediate XML file containing all the SQL commands required for an upgrade.

- Prompt you to start the Upgrade Wizard.

In some cases, you will need to modify the generated SQL files as required by Release Notes, Technical Notes, or Alerts before you run the Upgrade Wizard. To do this, answer No when prompted to run the Upgrade Wizard. Then, edit the SQL files and manually launch the Upgrade Wizard.

## How to Locate Master Configuration Files

Master upgrade configuration files are stored in the following location:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

Master upgrade configuration files use the following naming convention:

`master_UPGRADEOPTION_ENVIRONMENT_VERSION.ucf`

where:

- **UPGRADEOPTION** Indicates the upgrade process you are performing
  - *upgrep* Indicates Siebel database schema upgrade
  - *upgphys* Indicates Custom database schema upgrade indicates
  - *prepare\_for\_production\_upgrade* Indicates Prepare for production upgrade
- **ENVIRONMENT** Indicates the environment that you are upgrading
  - Development environment upgrades *dev*
  - Production environment upgrades *prod*
- **VERSION** Indicates the version from which you are upgrading

For example, if you are upgrading from Release 7.0.4, the master configuration file generated from the development environment upgrep is as follows:

`master_upgrep_dev_704.ucf`

The master configuration file generated from the Prepare for Production mode is as follows:

`master_prepare_for_production_upgrade.ucf`

### Related Topics

[“About the Siebel Upgrade Wizard and Driver Files” on page 77](#)

# About the Siebel Upgrade Wizard and Driver Files

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

The Upgrade Wizard performs the upgrade. It uses the information collected by the Database Server Configuration Utilities to execute commands and SQL scripts. These commands and SQL scripts are contained in driver files. Driver files are in ASCII text format and are organized into steps. The Upgrade Wizard reads the steps in the driver files and performs the commands contained in each step.

In a driver file, steps are separated by a blank line, and each step begins with a File Execute Entry number. The key part of each step is the command or script to be executed. When an SQL script is specified, you can review the script and see what changes it will make to the Siebel database before running the Upgrade Wizard.

Driver files are provided for each of the major upgrade operations. Examples of driver files are as follows:

- driver\_upgrep\_dev\_77.ucf
- driver\_upgphys\_dev\_77.ucf
- driver\_upgrep\_prod\_77.ucf
- driver\_upgphys\_prod\_77.ucf.

Here is an excerpt from a driver file that controls a development environment upgrep from Siebel 7.7 on Oracle. The excerpt contains three steps:

```
[Sql File Entry 0]
Type = DbSql
File Name = rename_existing_repositories.sql
Use Table Owner = 1
Use System Admin = 0
IgnoreAllDDLerrors = 0
IgnoreAllDMLerrors = 0
Argument 0 = $SiebelVersion
Title = Verify Repository Name
Title Message Num = 0
Estimated Disk Space = 0
Backup Db = 0
Parallelizable Item = 0
Prompt User For Status = 0
```

```
[File Execute Entry 1]
Type = FileExecute
File Name = $SiebelRoot\bin\odbcsql
Check Return Code = 1
Return Code Compliance = 0
16 Bit App = 0
```

```
Command Line = /s "$ODBCDataSource" /u $TableOwner /p $TablePassword /separator / /a /c rem /I  
$SiebelLogDir/dropif-db.log $DbsvrRoot/$DatabasePlatform/dropif-db.sql /v
```

Number of 10 Second Wait Loops = 2000

Return Code = 0

Title = Drop interface tables

Title Message Num = 0

Estimated Disk Space = 0

Backup Db = 0

Parallelizable Item = 0

Prompt User For Status = 0

[File Execute Entry 2]

Type = FileExecute

File Name = \$SiebelRoot\bin\odbcsql

Check Return Code = 1

Return Code Compliance = 0

16 Bit App = 0

```
Command Line = /s "$ODBCDataSource" /u $TableOwner /p $TablePassword /separator / /a /c rem /I  
$SiebelLogDir/trigdrop-db.log $DbsvrRoot/$DatabasePlatform/trigdrop-db.sql /v
```

Number of 10 Second Wait Loops = 2000

Return Code = 0

Title = Drop triggers

Title Message Num = 0

Estimated Disk Space = 0

Backup Db = 0

Parallelizable Item = 0

Prompt User For Status = 0

## How To Locate Upgrade Driver Files and SQL Scripts

Driver files are stored in the following location:

Windows: *DBSRVR\_ROOT\PLATFORM\UPGRADE\VERSION*

UNIX: *DBSRVR\_ROOT/bin/PLATFORM/UPGRADE/VERSION*

where:

- *PLATFORM* = the database type, for example ORACLE
- *VERSION* = the version from which you are upgrading, for example V7.7

For example, if you are upgrading from Siebel 7.7, the driver file for the development environment upgrep is as follows:

```
dri ver_upgrep_dev_77. ucf
```

Most of the SQL scripts generated or populated for the upgrade are stored in the same directory as the driver file. The remaining SQL scripts are stored in the `\PLATFORM` directory.

### Related Topics

[“About the Siebel Database Upgrade Process” on page 68](#)

[“About the Siebel Database Server Configuration Utilities” on page 72](#)

[“ForExample of a Siebel Development Upgrade Flow” on page 82](#)

[“About the Siebel Repository Merge” on page 89](#)

# About the Siebel Database Upgrade Log Files

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** Windows and UNIX only. Does not apply to IBM z/OS.

The Upgrade Wizard writes log files that provide detailed information on the upgrade processes, for example development-environment upgrep and upgphys. These log files provide detailed information on the upgrade steps, and they also list all errors. The Logparse utility analyzes and summarizes these log files.

## Upgrade Wizard Log Files

The Upgrade Wizard writes the log files for a process to the following directory:

Windows: `SIEBEL_ROOT\log\PROCESS`

UNIX: `$SIEBEL_ROOT/log/PROCESS`

where `PROCESS` defaults to `UPGRADEOPTI ON_ENVI RONMENT_VERSI ON`.

For example, the log files for a Siebel 7.7 development environment upgrep appear in the following locations:

Windows: `SIEBEL_ROOT\log\upgrep_dev_77`

UNIX: `$SIEBEL_ROOT/log/upgrep_dev_77`

You can select a different directory for the log files in the Log Output Directory screen in the Database Server Configuration Utilities.

Each subdirectory in the log directory contains the following files:

- **Output.** This subdirectory contains the Upgrade Wizard log files.
  - `upgwi z. log` (`srvrupgwi z. log` on UNIX). The Upgrade Wizard log for the process. This log contains detailed information on how the Upgrade Wizard ran. If you run the process more than once, the log name increments, for example `upgwi z2. log` or `srvrupgwi z2. log`.
  - Windows only. Log files with the name `upgwi zn(nnn)` can be ignored. For example, `upgwi z1(001). log`, `upgwi z1(002). log`, and so on.
  - Windows only. Log files with the name `sw_cfg_XXX` can be ignored. (These are located in the log directory.)
  - UNIX only. Log files with the name `srvrupgwi zn_nnn` can be ignored. For example, `srvrupgwi z1_01. log`, `srvrupgwi z1_02. log`, and so on.
  - Log files with the name `si ebel . log` can be ignored.

**TIP:** Each upgrade process, for example, a development upgprep, has a driver file located in `\dbsrvr\platform\upgrade\version`. An example of a driver file: `driver_upgprep_dev_77.ucf`. The Upgrade Wizard performs the steps in the driver file. If a step calls a utility, such as `ddlmp` or `dataimp`, the `/l` option in the command syntax specifies the name of the log file. Unless you modified the output location for the log, `$Siebel LogDir` is `\siebsrvr\log\process`.

- **State.** This subdirectory contains the state.log file.
  - **state.log.** The state.log file lists each step in the upgrade process and whether the step completed successfully.

The output and state log file directories are automatically archived on subsequent runs of a process that completes successfully. (The names of subsequent log directories are appended with `_1`, `_2`, and so on.) To preserve disk space, periodically delete or save log directories to another location.
- **Summary.** This subdirectory is generated by the Logparse utility. The Logparse utility also generates a report that summarizes the upgrade, `summary.html`. The `summary.html` file contains links to the html pages in the summary subdirectory.
- **Summary.html.** This file is created by the Logparse utility. It summarizes the Upgrade Wizard log files in the output directory. Use this file to verify that all upgrade steps completed successfully and to determine if there were any errors.
- **Summary.txt.** Text file created by the Logparse utility. This file contains the same information as `summary.html`.

**NOTE:** If a browser is not installed on the machine you are using, review `summary.txt` instead of `summary.html`.
- **Summary.xml.** File produced by the Logparse utility during production upgrades and used by the Siebel Upgrade Tuner to parallelize table creation and index creation and to inactivate SQLs that affect no rows. (Development upgrades do not produce a `summary.xml` file.)
- **Upgtuner\_ftp\_get.txt. (UNIX Only).** File produced by the Logparse utility during production upgrades. You can use this file to transfer the upgrade scripts from your UNIX machine to a Windows temporary directory to perform upgrade tuning. (Development upgrades do not produce `upgtuner_ftp_get.txt`.)

- **Upgtuner\_ftp\_put.txt. (UNIX Only).** File produced by the Logparse utility during production upgrades. After you tune your production upgrade scripts, you can use this file to transfer the upgrade scripts from the temporary directory on your Windows machine back to your UNIX environment. (Development upgrades do not produce upgtuner\_ftp\_put.txt.)

### The Logparse Utility

For each upgrade process, the Logparse utility analyzes the Upgrade Wizard log files in `si_ebsrvr\log\process`. It then summarizes the log files and provides several reports. The Logparse reports provide the following information:

#### Parameters report

The Parameters report lists the total time required for the upgrade process. It also lists environment variables and input files used by the upgrade process. Use this report to verify that the upgrade was set up correctly.

#### Step/Error Summary

The Step/Error Summary lists each step in the upgrade process and whether it completed successfully. It also provides a link to a detailed step summary for each step.

The detailed step summary lists database errors returned by the Upgrade Wizard and by SQL scripts. It does not list errors returned by upgrade utilities called by the driver files, such as `ddlmp` and `dataimp`. You must review the upgrade logs for these utilities separately.

The detailed step summary lists all errors found in the log files, including benign errors. An errors file (`errors.xls`, `errors.rtf`, and `errors.txt`) is provided that lists benign errors for each database type. When you find an error in the `summary.html` file, you must compare it to the errors file to determine if it is benign.

#### ■ Performance report

The performance report provides a list of the longest running SQL scripts and a list of SQL scripts returning 0 rows. The lists include a link to each script. Upgrade Tuner uses this information as input to help you tune upgrade performance.

The report also provides two lists of DDL commands executed during the upgrade process. The DDL Summary lists the tables that were created, altered, or deleted by the `ddlmp` utility. The Native DDL Summary lists tables that were created, altered, or deleted by SQL scripts and includes a link to each script.

**Platforms:** MS Windows, UNIX, IBM z/OS.

## For **Example of a Siebel Development Upgrade Flow**

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

This topic presents the flow of steps in part of a typical development environment upgrade. The steps are extracted from an actual driver file. To perform an upgrade, the Upgrade Wizard reads the steps in a driver file and performs the commands the steps contain.

The driver file type used in this example is as follows:

- Upgrade: Siebel Industry Application (SIA) 7.7 to Siebel SIA 7.8
- Environment: development
- Upgrade mode: upgrep
- Database: Oracle
- Multilingual: No

Table 25 lists the steps in the driver file. Key steps are in bold.

The Script or Input File column in the table lists the SQL file or input file that is executed in each step. The Comment column provides a brief explanation of what the SQL file or input file does.

The SQL files used for an upgrade and the contents of the SQL files vary depending on the upgrade mode and database.

Table 25. Example of Steps in a Development Environment Upgrep

Step	Script or Input File	Comment
Determine collation sequence of database.	Not applicable	Determines database sort order.
Verify repository name	rename_existing_repositories.sql	Renames Siebel Repository to Prior Customer Repository.
Drop interface tables	dropif-db.sql	Drops all EIM tables.
Drop database triggers	trigdrop-db.sql	Drops all dynamically created triggers.
Drop database-level functions and procedures	drop_db_func_proc.sql	Drops the exchange rate function exrate.
<b>Prepare for table creation</b>	pret.sql	Drops specified tables. Performs DDL operations such as adding columns to tables. Performs DML operations such as revising date formats.
Create temporary tables for SIA	<ul style="list-style-type: none"> <li>■ ddlimp utility</li> <li>■ ddl_temp_sia.ctl as input</li> </ul>	The input file specifies the structure of the tables to be created or updated. These tables are used to perform data migration and other DML changes.

Table 25. Example of Steps in a Development Environment Upgrep

Step	Script or Input File	Comment
Prepare for table creation for SIA	pret_sia.sql	Drops specified tables. Performs DDL operations such as adding columns to tables. Performs DML operations such as revising date formats.
<b>Create and update tables</b>	<ul style="list-style-type: none"> <li>■ ddlimp utility</li> <li>■ ddl.ctl as input</li> </ul>	The ddl.ctl file specifies the structure of tables to be created or updated.
Create temporary tables for stored procedures	<ul style="list-style-type: none"> <li>■ ddlimp utility</li> <li>■ ddlsptbl.ctl as input</li> </ul>	The input file specifies the structure of temporary tables to be created or updated.
Household data migration for FINS	Household_Mig_Fins.sql	Creates and populates specified temporary tables. Then migrates data to them and performs DML operations. Migrates data back to primary tables. Drops temporary tables.
<b>Prepare for index creation</b>	preschem.sql	Performs DML operations. Moves data between tables. Changes data in existing fields based on specified conditions.
Prepare for index creation for SIA	preschem_sia.sql	Same as preschem.sql.
<b>Create indexes</b>	<ul style="list-style-type: none"> <li>■ ddlimp utility</li> <li>■ ddl.ctl as input</li> </ul>	The input file specifies the structure of indexes to be created.
Delete old license key	delappkey.sql	Deletes the Siebel license key from S_APP_KEY.

Table 25. Example of Steps in a Development Environment Upgrep

Step	Script or Input File	Comment
<b>Import seed data</b>	<ul style="list-style-type: none"> <li>■ dataimp utility</li> <li>■ seedupg0.inp as input</li> <li>■ seedupg1.inp as input</li> <li>■ seedupg_locale.inp as input</li> </ul>	<p>Prior to importing seed data, dataimp deletes existing seed data.</p> <p>The seedupg* files contain filters that dataimp uses to prevent deleting seed data you have modified or seed data meeting specified criteria.</p> <p>Unmodified seed data has a last update date (LAST_UPD) of 1980-01-01. Dataimp does not delete records where LAST_UPD is later than this date.</p>
Upgrade data after seed data import	upg_data_afterseed.sql	For customers who have not converted to UTC time, sets the UTC value in S_SYS_PREF to False. For customers who have converted to UTC time, the script takes no action.
Upgrade data after seed data import SIA	upg_data_afterseed_sia.sql	Empty.
Set system preference for codepage for DB	set_codepage.sql	Sets the database codepage in S_SYS_PREF.
Set system preference for Unicode codepage for DB	set_unicode.sql	Sets the Unicode codepage to UTF-8 in S_SYS_PREF
Update version component information	upd_upgcomp.sql	<p>Updates S_UPG_COMP with the product release level. Also updates location of srf file.</p> <p>S_UPG_COMP stores version information for application executables.</p>
Run Oracle-specific DDL commands	ddlora.sql	Creates Oracle-specific DDL information, such as default storage parameters for docking objects, repository objects, and seed objects.

Table 25. Example of Steps in a Development Environment Upgrop

Step	Script or Input File	Comment
<b>Import common ancestor repository</b>	<ul style="list-style-type: none"> <li>■ repimexp utility</li> <li>■ Standard Siebel Repository as input</li> </ul>	Imports the Standard Siebel Repository into S_REPOSITORY. For example, if you are upgrading from Siebel 7.5.3, this command imports the standard Siebel 7.5.3 repository.
Rename EIM columns/indexes	rmv_anc_eim_proc_col_ind.sql	<p>Removes EIM processing columns and indexes from the Prior Customer Repository and the common ancestor repository (Standard Siebel Repository). This prevents the repository merge from preserving EIM columns incorrectly.</p> <p>The merge will preserve only those Siebel EIM columns shipped with the new Release.</p>
Update Siebel database version	<ul style="list-style-type: none"> <li>■ update_ver.sql</li> <li>■ seeduver.sql</li> </ul>	The update_ver.sql script creates a temporary table, S_APP_VER_TEMP, which contains new version information for the database schema. The seeduver.sql script updates S_APP_VER with this information.
<b>Import new Siebel Repository</b>	<ul style="list-style-type: none"> <li>■ repimexp utility</li> <li>■ Imports New Siebel Repository</li> </ul>	<p>Imports the New Siebel Repository into S_REPOSITORY.</p> <p>Revises schema version information in S_APP_VER.</p>
<b>Import New Customer Repository</b>	<ul style="list-style-type: none"> <li>■ repimexp utility</li> <li>■ Imports New Customer Repository</li> </ul>	<p>Imports the New Customer Repository into S_REPOSITORY.</p> <p>Revises schema version information in S_APP_VER.</p>
Restore database version	restore_ver.sql	Uses S_APP_VER_TEMP to update the schema version information in S_APP_VER. Drops S_APP_VER_TEMP.

Table 25. Example of Steps in a Development Environment Upgrade

Step	Script or Input File	Comment
Upgrade repository data SIA	repos_upgrade_sia.sql	Empty.
<b>Upgrade repository data</b>	repos_upgrade.sql	Makes specific repository-related changes to repository records and to other tables.
Set repository workflow domains to MLOV	set_multilingual.sql	Empty.

Table 25. Example of Steps in a Development Environment Upgrep

Step	Script or Input File	Comment
Install SQL packages	<ul style="list-style-type: none"> <li>■ seeduver.sql</li> <li>■ ifstrg.sql</li> <li>■ ifindxstrg.sql</li> <li>■ pkgseq.sql</li> <li>■ pkgldel.sql</li> <li>■ trgreset.sql</li> <li>■ ddlseq.sql</li> <li>■ pkgvis.sql</li> <li>■ delete_dock_rules.sql</li> </ul>	<ul style="list-style-type: none"> <li>■ <b>seeduver.sql.</b> Verifies that versions are set correctly in S_APP_VER.</li> <li>■ <b>ifstrg.sql.</b> Sets storage parameters for EIM tables.</li> <li>■ <b>ifindxstrg.sql.</b> Sets storage parameters for EIM table indexes.</li> <li>■ <b>pkgseq.sql.</b> Adds a suffix to row IDs in the S_SEQUENCE table. Ensures that row IDs are unique.</li> <li>■ <b>pkgldel.sql.</b> Defines s_txn_log_del_proc. Procedure periodically deletes transactions from S_DOCK_TXN_LOG. Also deletes rows from S_DOCK_TXN SET. Prevents need for large rollback segment.</li> <li>■ <b>trgreset.sql.</b> Ensures that denormalized rows in S_TERR have correct values.</li> <li>■ <b>ddlseq.sql.</b> Sets sequence numbers for specified tables.</li> <li>■ <b>pkgvis.sql.</b> Creates function that modifies how Oracle optimizer does visibility check.</li> <li>■ <b>delete_dock_rules.sql.</b> Deletes Prior Customer Repository routing rules from S_DOC_VIS_RULE that meet specified conditions. Attempts to preserve rules added using Docking Wizard.</li> </ul>

Table 25. Example of Steps in a Development Environment Upgrep

Step	Script or Input File	Comment
Create database-level functions and procedures	db_func_proc.sql	Creates or replaces the currency exchange rate function: exrate.
Upgrade ISS data	upg_iss.sql	Updates data stored in tables that support Oracle's Siebel Product Configurator.
<b>Set primary children in data tables</b>	<ul style="list-style-type: none"> <li>■ gen_primary1.sql</li> <li>■ gen_primary2.sql</li> <li>■ gen_primary3.sql</li> <li>■ gen_primary4.sql</li> </ul>	<ul style="list-style-type: none"> <li>■ <b>gen_primary1.sql</b>. Sets primary child for S_DOC_QUOTE</li> <li>■ <b>gen_primary2.sql</b>. Empty.</li> <li>■ <b>gen_primary3.sql</b>. Sets primary child for S_LOY_PROMO</li> <li>■ <b>gen_primary4.sql</b>. Empty</li> </ul>

**Related Topics**

["About the Siebel Database Server Configuration Utilities" on page 72](#)

["About the Siebel Upgrade Wizard and Driver Files" on page 77](#)

## About the Siebel Repository Merge

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Repositories are located in tables in the Siebel database. These tables store the definitions of the objects used to build Siebel Applications. These tables also store Siebel schema definitions. When you display objects in the Siebel Tools Object List Explorer, you are displaying records from these tables.

The repository tables contain two types of records:

- Object definitions used to create Siebel applications, such as business components, applets, controls, and the relationships between them.
- Definitions of the tables in the Siebel database (metadata). These records define the logical schema of the Siebel database. Later in the upgrade process, you will synchronize the physical schema of the Siebel database with the logical schema as defined by these records.

Repository records include a repository ID that identifies the repository to which the record belongs. The repository ID forms part of the user index for records and allows several repositories to be stored in the same set of tables.

The Siebel Repository is the deployed, active Tools repository. You use this repository to customize applications and create SRF files for deployment.

The development upprep adds three additional repositories to the Siebel database, as shown in [Table 26](#). These repositories are required for the repository merge.

Table 26. Development Database Repositories

Repository Name	Added During Upgrade?	Siebel Tools Name	Description
Prior Customer Repository	No	Prior Customized	This is your current Siebel Repository. It contains any changes you have made. You renamed it Prior Customer Repository before doing the initial database upgrade.
Prior V6.x Siebel Repository or Prior V7.x Siebel Repository	Yes	Prior Standard	The standard Siebel Repository for your installed release (the one you are upgrading from).  Is called the common ancestor repository in the upgrade SQL scripts.
New Siebel Repository	Yes	New Standard	The Siebel Repository for the release to which you are upgrading. This repository defines the new release.
New Customer Repository	Yes	New Customized	A second copy of the New Siebel Repository. Your customizations from the Prior Customer Repository are merged into this repository. After the upgrade, this becomes the Siebel Repository.

### What Happens During a Repository Merge?

The repository merge process identifies differences between the repository in your old release (Prior Customer Repository) and the repository in the new release (New Siebel Repository). The merge process then merges these differences into the New Customer Repository. The merge process searches for the following types of objects in the Prior Customer Repository: customer-created, customer-deleted, and customer-modified.

### Customer-Created Objects

Customer-Created objects are top-level objects, such as screens, applets, and business components, that you have created in the Prior Customer Repository. If an object is present in the Prior Customer Repository but not the Prior Siebel Repository, it is customer-created. The merge process copies customer-created objects intact to the New Customer Repository.

### Customer-Deleted Objects

Customer-Deleted objects are top-level objects you have deleted in the Prior Customer Repository. If an object is absent in the Prior Customer Repository but present in the Prior Siebel Repository, it is customer-deleted.

If you delete a top-level object in the Prior Customer Repository and it is present in the New Customer Repository, the merge process does not delete the object from the New Customer Repository. After the merge, you must review these objects and remove them as desired.

If you delete a top-level object from the Prior Customer Repository, and it is obsolete (inactive) in the New Customer Repository, the object will be present and inactive in the New Customer Repository.

### Customer-Modified Objects

A customer-modified object has the following characteristics:

- It is a top-level object, such as screen, applet, or business component.
- The object exists in the Prior Siebel Repository and the Prior Customer Repository. (The object is not customer-created or customer-deleted)
- The properties of the object in the Prior Customer Repository and Prior Siebel Repository are not the same. (You have modified the object).

If the object properties are also different between the Prior Siebel Repository and New Siebel Repository, the object has changed in the new release, and a conflict exists. The merge process logs the conflict and resolves it. After the merge, you must review how these conflicts were resolved and change the resolutions as desired.

For customer-modified objects where no conflict exists, the merge process copies the modifications to the object to the New Customer Repository.

If you modify a top-level object by deleting any of its child objects, the merge process does not delete these child objects in the New Customer Repository. After the merge, you must review child objects and remove them as desired.

Some of the child objects of Applet are an exception. If you delete the following child objects of Applet from the Prior Customer Repository, the merge process deletes these objects from the applet in the New Customer Repository:

- Control
- List Column
- Page Tab
- Chart

## How the Siebel Database Upgrade Works ■

- Applet Web Template Item
- View Web Template Item

For example, if you delete a button from Applet-A in the Prior Customer Repository. The merge process deletes this button from Applet-A in the New Customer Repository.

Table 28 shows how a regular merge handles both customer-modified and customer-created objects. The columns list the status of a repository object:

- **Standard.** The object appears in the Prior Standard Repository, in the New Siebel Repository, and is not customer-modified.
- **Deleted.** You have deleted the object from the Prior Customer Repository (customer-deleted).
- **Customized.** You have modified the object in the Prior Customer Repository (customer-modified).
- **Revised.** The object has changed in the new release (New Siebel Repository).
- **New.** You have created the object in the Prior Customer Repository (customer-created), or the object is new in the new release (New Siebel Repository).
- **Inactive.** The object is present in the New Siebel Repository and New Customer Repository but is inactive and not used in the new release. The object is obsolete.

The first three columns list the status of the object in the three repositories that the merge process compares during the merge. The last column, Merged New Customer Repository, lists the status of all top-level repository object types after the repository merge is complete and the postmerge utilities have run.

Table 27. How the Merge Handles Repository Objects

Prior Standard Repository (PSR)	Prior Customer Repository (PCR)	New Siebel Repository (NSR)	Merged New Customer Repository
Standard	Standard	Standard	Standard
Standard	Standard	Revised	Revised
Standard	Standard	Standard/ Inactive	Standard/Inactive
Standard	Customized	Standard	Customized
Standard	Customized	Standard/ Inactive	Customized/Inactive
Standard	Customized	Revised	Revised (conflict)
Not applicable	New	Not applicable	New
Not applicable	Not applicable	New	New
Standard	Deleted	Standard	Standard

Table 27. How the Merge Handles Repository Objects

Prior Standard Repository (PSR)	Prior Customer Repository (PCR)	New Siebel Repository (NSR)	Merged New Customer Repository
Standard	Deleted	Standard/ Inactive	Standard/Inactive
Standard	Deleted	Revised	Revised

### Upgrade Ancestor Property

You can link repository objects together so that one object inherits the upgrade behavior of another. You do this by specifying an upgrade ancestor for an object. You can specify upgrade ancestors for the following object types:

- Applets
- Business Components
- Integration Objects
- Reports

For example, you create Applet-B by copying Oracle standard Applet-A. In Applet-B you specify Applet-A as the upgrade ancestor. In the New Siebel Repository, Applet-A has a new button. Because Applet-B is a descendant of Applet-A, the merge process adds the new button to both Applet-A and Applet-B.

If you had not specified Applet-A as an upgrade ancestor of Applet-B, Applet-B would not have received the new button.

### Postmerge Utilities

The postmerge utilities run after the repository merge completes. The utilities make changes to objects in the New Customer Repository. The purpose of the postmerge utilities is to reduce the configuration changes required as part of reviewing applications and UI after the merge.

### Incorporate Custom Layout

If you are upgrading from Siebel 7.x or later, you can choose Incorporate Custom Layout (ICL) when performing a repository merge. ICL is intended to reduce postmerge UI reconfiguration workload for customer-modified UI objects.

An ICL merge is a regular merge that contains two extra steps. The first step occurs at the end of the merge and applies special handling to screens, views, and applets. The second step is part of the postmerge utilities and provides specialized web template files similar to the release you are upgrading from.

### Additional Merge Steps if You Are Upgrading from Siebel 6.x

For upgrades from Siebel 6.x, the repository merge process includes several additional steps. These steps convert the Windows-based UI to HTML and transition scripts to the Siebel 7.x architecture:

- 1 Prepare the Prior Customer Repository. A wizard does the following:
  - Migrates strings from S\_MSG to locale tables.
  - Merges labels and fields. In Siebel 7.x, labels are a caption property of controls.
  - Merges Web templates. Siebel 7.x uses a revised method for providing base, edit, query, and new record functionality in applet Web templates. The wizard revises applet templates to use the new method.
- 2 Run the repository merge and postmerge utilities.
- 3 Run the Web Client Migration Wizard. This merges applets and views that you have customized or created into the New Customer Repository. You do this by assigning model applets and model views to them.
- 4 Manually, revise and reassign server and browser scripts.
- 5 Manually, revise any integrations with third-party products.

#### Related Topics

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

[“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94](#)[“About the Siebel Postmerge Utilities” on page 97](#)

[“Reviewing Siebel Repository Object Property Conflicts” on page 238](#)

## About Inheriting Upgrade Behavior in a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

You can link objects together so that one object inherits the upgrade behavior of another. You do this by specifying an upgrade ancestor for an object.

Use standard objects as upgrade ancestors. A **standard object** is an uncustomized repository object provided by Oracle.

A common customization strategy is to create new objects by making a copy of a standard object and then modifying the copy, called the **descendant**.

For descendants that are the following object types, you can specify an upgrade ancestor:

- Applets
- Business Components

- Integration Objects
- Reports

You specify the upgrade ancestor in the descendant's Upgrade Ancestor field in Siebel Tools. During the repository merge, the descendant is upgraded in the same way as the upgrade ancestor. (Upgrade ancestors of applets are ignored in an ICL merge.)

For example, you create Applet-B by copying standard Applet-A. In Applet-B you specify Applet-A as the upgrade ancestor. In the New Siebel Repository, Applet-A has a new button.

After a regular merge, both Applet-A and Applet-B will have the new button.

After an ICL merge, Applet-B will have the new button, but Applet-A will not:

- During the regular merge, Applet-B receives the new button because it is a descendant of Applet-A.
- During the ICL merge portion, the web template configuration of Applet-A is copied from the Prior Customer Repository to the New Customer Repository. This web template configuration does not contain the new button. This removes the new button from Applet-A.
- Since the ICL merge portion ignores the Upgrade Ancestor property, this change is not propagated to Applet-B. Applet-B retains the new button.

## Limitations on the Upgrade Ancestor Property

The Upgrade Ancestor property is considered only during repository merges as part of application upgrades under these conditions:

- If an upgrade ancestor is inactive in the New Siebel Repository, it is obsolete, and its upgrade behavior is not propagated to descendants.  
If an ancestor object is obsolete in the New Siebel Repository, descendants are not also obsolete.  
If an upgrade ancestor is not present in the New Siebel Repository, error messages display during the repository merge and are written to the merge log file. These errors are acceptable and do not mean the merge has failed.
- The Upgrade Ancestor property is not considered during repository imports. However, imported objects can specify an upgrade ancestor. When the next application upgrade is done, the Upgrade Ancestor property is taken into account.
- The setting of the Upgrade Ancestor property is not considered when applying application patches. If the upgrade ancestor is modified by the patch, descendants are not modified.
- Use caution when specifying upgrade ancestors. For regular merges, setting the Upgrade Ancestor property on applets propagates merge problems from standard-object applets to descendant applets.
- Specifying an upgrade ancestor for objects slows the repository merge.

## Upgrade Ancestor Picklist

When you click in the Upgrade Ancestor field, a picklist displays. The following criteria are used to populate the picklist:

### Applets

- Table is the same as the current applet buscomp
- Class is the same as the current applet class
- Upgrade Ancestor is null
- Applet is a standard object

### Reports

- Buscomp is the same as the current report buscomp
- Class is the same as the current report class
- Upgrade Ancestor is null
- Report is a standard object

### Business Components

- Bus Comp is the same as the current business component
- Class is the same as the current business component
- Upgrade Ancestor is null
- Business component is a standard object

### Integration Objects

- 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
- Integration object is a standard object

## Propagating Changes to Objects After the Merge

If you do not select an upgrade ancestor for an object, changes to the upgrade ancestor are not propagated to the descendant during the repository merge.

You can manually propagate changes to descendants after the merge by using the Siebel Tools object comparison and synchronization features. These features enable you to compare any two objects and propagate differences to one or both of the objects. For more information, see *Configuring Siebel Business Applications*.

### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## About the Siebel Postmerge Utilities

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

The postmerge utilities start automatically after the merge is complete. The postmerge utilities make revisions to objects in the New Customer Repository. The utilities are intended primarily to identify UI configuration problems for upgrades prior to Siebel 7.7. (At Siebel 7.7, there were significant changes to UI navigation.) The utilities are also useful for identifying problems with UI customizations for upgrades after Siebel 7.7.

The postmerge utilities are implemented as a framework. Each utility in the framework is a business service and is listed in `reputi l i ty. xml` :

Windows: `SIEBEL_TOOLS_INSTALL_DIR\reppatch\reputi l i ty. xml`

Where

`SIEBEL_TOOLS_INSTALL_DIR` is the directory where Siebel Tools is installed.

The postmerge utilities are as follows:

- **CSSGridRepPatch**. Converts customized applet controls to grid-based layout.
- **CSSUI NavUpgradeReposPrep**. Converts customized screens to the Siebel 7.7 UI navigation scheme.
- **CSSMVGUpgradePatch77**. Enables the shuttle feature in customized MVGs.
- **CSSUI UpgradeReports**. Lists unresolved problems in converting customized screens and views to the Siebel 7.7 UI navigation scheme.
- **CSSWebTemplatePatch**. For ICL upgrades, the utility copies Web templates into Siebel Tools that support the release you are upgrading from.
- **CSSWFRepPatch**. For workflows, this utility changes step references from row-id references to name references for child objects of steps.

### CSSGridRepPatch

As of Siebel 7.7, the layout of form applets is grid-based, rather than flow-based. Applet Form Web templates have several new properties: Grid Property, Row Span, and Col Span. In addition, item identifier syntax changes to `xxyy`, where `xx` is the grid row and `yy` is the grid column where the control is located.

**Regular merge.** The utility identifies both customer-created and customer-modified applets and adjusts the properties of controls in them as follows:

- If you have modified the location of an existing control, the utility restores the control to its original location. If you have modified other properties of the control, these changes are preserved. You must move the control and label to the desired location.

- If you have added a new control, it does not display after the merge. To display the control, the utility assigns an item identifier that places the control in the lower left portion of the applet. The utility creates a Grid Property, Row Span, and Col Span property for the item. The utility also creates a Label control. You must move the new control and label to the desired location.
- The utility does not change property settings for controls that are unmodified.

**ICL merge.** (Applies only to applets where Upgrade Behavior is not set to Admin.) For upgrades from releases prior to Siebel 7.7, form applets copied-in from the Prior Customer Repository remain flow-based and display normally. The utility does not convert these applets to grid-based. New and modified controls display in the correct locations. For upgrades from Siebel 7.7 or later, the utility makes no changes.

The utility generates output to the `reputility.log` section called `Invalid Applet Web Template Item Mapping Cleanup`.

### CSSUI NavUpgradeReposPrep

At Siebel 7.7, a declarative model for associating views with screens was introduced. The relationship between views and screens for all levels of navigation must be explicitly declared. Several new object properties for screens and views are introduced to support this.

The new navigation scheme is applied to all screens and views for both regular and ICL merges. The utility runs after the repository merge in all upgrades. However, it is intended primarily for upgrades from releases prior to Siebel 7.7.

After a regular or ICL merge, screens and views display as follows:

- Regular merge.
  - Unmodified screens display normally.
  - The views in customer-created screens do not display.
  - Views added to existing screens do not display.
- ICL Merge
  - Unmodified screens with Upgrade Behavior property set to Admin display normally.
  - Views in unmodified screens with Upgrade Behavior set to Preserve or NULL do not display.
  - Views in customer-created screens do not display.
  - Views added to existing screens do not display, regardless of Upgrade Behavior setting.

The utility scans all screen view definitions looking for orphaned views. It groups orphaned views under existing Categories. If no Category exists, the utility creates one. This causes the orphaned views to display.

For a regular merge, you must review customer-created and customer-modified screens and views to verify that views are correctly associated with screens.

For an ICL merge, you must also review these screens and views. In addition, you must review all screens and views where Upgrade Behavior is set to Preserve or NULL.

The utility generates output to the `reputility.log` section called `User Interface Navigation Upgrade`.

## CSSMVGUpgradePatch77

As of Siebel 7.7, MVG applets with an M:M relationship to the underlying business component are configured as shuttle applets by default. The utility scans these MVG applets in the New Customer Repository and reconfigures them to display as shuttle applets:

- **Regular merge.** After the merge, customer-created and customer-modified MVGs are not shuttle-enabled. The utility reconfigures these MVGs so they display as shuttle applets.
- **ICL merge.** After an ICL merge, the utility reconfigures MVGs as follows:
  - Customer-created and customer-modified MVGs are not shuttle-enabled. The utility reconfigures these MVGs so they display as shuttle applets.
  - Unmodified MVGs that have Upgrade Behavior set to Preserve or NULL are copied to the New Customer Repository during the merge and replace MVGs that are shuttle-enabled. The copied-in MVGs may not be configured as shuttle applets. The utility reconfigures the copied-in MVGs so they display as shuttle applets.
  - Unmodified MVGs that have Upgrade Behavior set to Admin are not inactivated in the New Customer Repository and so display as shuttle applets.

The utility generates output to the `reputi li ty. log` section called Multi Value Group Shuttle Applet Upgrade.

## CSSUIUpgradeReports

This utility makes no changes to the New Customer Repository. Instead, it scans UI objects in the repository and lists problems that could not be resolved by the `CSSUINavUpgradeReposPrep` utility.

The `CSSUIUpgradeReports` utility writes the report to `reputi li ty. log`. You must manually correct problems listed in the report. The report is located in the POST MERGE USER INTERFACE REPORTING UTILITY section of the log.

The report has the following sections:

- Issue 1: Rich Text Control (RTC) that needs to have User Properties Reconfigured
- Issue 2: New Aggregate Category Records that should be renamed
- Issue 3: Views that need an applet in View Web Template Item Id 1
- Issue 4: Chart Views Needing Migration to Aggregate Type
- Issue 5: Explorer Views Needing Migration to Aggregate Type
- Issue 6: Categories where parent applets are missing drilldowns to a Detail View

## CSSWebTemplatePatch

This utility runs after an ICL merge. It does not run after a regular merge. The utility performs the following steps in the installation directory of the Siebel Tools that you used to perform the merge:

- The utility moves the Web template files for the new release from `webtempl` to `\temp\webtempl`.

## How the Siebel Database Upgrade Works ■

- The utility copies Web template files from a subdirectory of repatch\web\_templates to \webtempl.

These Web template files support the UI of the release you are upgrading from. They are very similar but not identical to the Web template files included in that release.

For example, if you selected the 7.5.3 and “Label on Top” ICL options, the utility copies Web template files from the 753 and TopLabel subdirectories of repatch.

The utility generates output to the reproductibility.log section called Web Template PCL Patch Rule.

### CSSWFRepPatch

This utility changes workflow step references from row-id references to name references for child objects of steps. This completes the process of migrating workflows to the repository at Siebel 7.7. This utility primarily affects upgrades from releases prior to Siebel 7.7.

This utility does not write to the reproductibility.log.

### How the Postmerge Utilities Work the Upgrade Behavior Property

The postmerge utilities ignore the Upgrade Behavior property. They make changes to UI objects based on the object’s characteristics rather than Upgrade Behavior setting.

#### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

[“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## About the Siebel Incorporate Custom Layout (ICL) Upgrade Option

**Upgrades from:** Siebel 7.x using ICL.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

In each new release, there are typically several types of changes to existing UI objects:

- Screen layouts are revised
- View layouts are revised
- Controls are added or removed in applets
- Controls are moved in applets

If you have modified existing screens, views, and applets, the changes in the new release can require significant UI layout reconfiguration after the repository merge. The purpose of an ICL merge is to reduce postmerge UI layout reconfiguration.

## Overview of Incorporate Custom Layout

If you are upgrading from Siebel 7.x or later, you can choose Incorporate Custom Layout (ICL) when performing a repository merge. An ICL merge handles customer-modified screens, views, and applets differently than other repository objects.

A regular merge identifies changes you have made to objects in the Prior Customer Repository (customer-modified objects) and merges those changes into the counterpart objects in the New Customer Repository. For example, if you added a control to Applet-A in the Prior Customer Repository, a regular merge adds the control to Applet-A in the New Customer Repository.

An ICL merge does not merge changes into counterpart objects in the New Customer Repository. Instead, an ICL merge process replaces the configuration of screens, applets, and views in the New Customer Repository with those from the Prior Customer Repository.

For example, you have added a control to Applet-A in the Prior Customer Repository. The merge process inactivates the Web template configuration for Applet-A in the New Customer Repository and replaces it with the Web template configuration for Applet-A from the Prior Customer Repository. The Web template configuration from the Prior Customer Repository contains the new control along with the correct layout of the control.

In addition, after the merge, the postmerge utilities copy-in new Web template files that are similar to the release you are upgrading from. Web template files provide page containers that control screen, view, and applet layouts. The copied-in Web template files further ensure that the UI in the new release has layouts similar to the UI in the old release.

An ICL merge adds two additional steps to the regular merge process. These two steps are called *ICL handling*:

- At the end of the merge process, Screen, View, and Applet object types along with specific child objects, are copied to the New Customer Repository from the Prior Customer Repository. They replace their counterpart objects in the New Customer Repository.
- When the postmerge utilities run, the Web template files for the new release are copied out and saved. New Web template files similar to those of the release you are upgrading from are copied-in.

Customer-created screens, views, and applets do not receive ICL handling. These UI objects are copied to the New Customer Repository intact. This is true even if a customer-created applet specifies an ICL object applet as an upgrade ancestor.

## Who Should Use ICL?

Consider using ICL if it can significantly reduce your postmerge UI cleanup workload. ICL merges are intended for customer-modified UI objects. A *customer-modified UI object* is a standard screen, view, or applet you have modified and saved without changing its name. Screens, views, and applets that you create by copying them from a standard object and renaming them are *customer-created UI objects*.

Use the following factors to assess whether to use ICL:

- ICL can be used only for upgrades from Siebel 7.x. If you are upgrading from Siebel 6.x to 7.7, you cannot use ICL.

- If your UI customizations consist mostly of customer-modified objects, then ICL is a good choice. ICL is intended to preserve the layout and contents of customer-modified screens, views, and applets.
- If your customizations consist primarily of customer-created UI objects that specify an upgrade ancestor, ICL is not recommended. This is because ICL ignores the Upgrade Ancestor property. For UI objects, if the configuration of an upgrade ancestor changes in the new release, the changes are not propagated to the descendant.
- In each release, ICL handling cannot be applied to screens, views, and applets that have Upgrade Behavior set to Admin in Siebel Tools. If you have large numbers of customer-modified screens, views and applets that are excluded from ICL handling, ICL may not be a good choice.
- You can select ICL for only every other upgrade. Choose ICL for upgrades where the most customizations occur.

At the upgrade after an ICL upgrade, you must bring UI layouts forward to those of the installed release before performing the repository merge.

### UI Objects That ICL Affects

To receive ICL handling, a UI object must meet all the following criteria. Objects that meet all the criteria are called *ICL objects*:

The object must be one of the following top-level object types. For each top-level object type, the child objects that ICL affects are also listed:

- Screen
  - Screen View and all child objects
- View
  - View Web Template and all child objects
- Applet
  - Applet Web Template and all child objects
  - Drilldown and all child objects
- Screens, Views, and Applets, must be a standard object or a customer-modified object. (Customer-Created screens, views, and applets receive regular merge handling.)
- The object must be *preservable*. A screen, view, or applet is preservable if its Upgrade Behavior property is not set to Admin.

In summary, both customer-modified and standard unmodified screens views, and applets that are preservable receive ICL handling.

[Table 28](#) compares a regular merge and an ICL merge. The columns list the status of a repository object:

- **Standard.** The object appears in the Prior Standard Repository, in the New Siebel Repository, and is not customer-modified.
- **Deleted.** You have deleted the object from the Prior Customer Repository (customer-deleted).

- **Customized.** You have modified the object in the Prior Customer Repository (customer-modified).
- **Revised.** The object has changed in the new release (New Siebel Repository).
- **New.** You have created the object in the Prior Customer Repository (customer-created), or the object is new in the new release (New Siebel Repository).
- **Inactive.** The object is present in the New Siebel Repository and New Customer Repository but is inactive and not used in the new release. The object is obsolete.

The first three columns list the status of the object in the three repositories that the merge process compares during the merge. The last two columns list the status of the object after the merge is complete and the postmerge utilities have run.

The Merged New Customer Repository column lists the status of all repository object types after the repository merge.

The ICL Merged New Customer Repository column lists the status of screen, view, and applet ICL objects after an ICL merge.

Exceptions listed in [Table 28](#) are described in a section following the table.

Table 28. How the Merge Handles Repository Objects

Prior Standard Repository (PSR)	Prior Customer Repository (PCR)	New Siebel Repository (NSR)	Merged New Customer Repository	ICL Merged New Customer Repository
Standard	Standard	Standard	Standard	Standard
Standard	Standard	Revised	Revised	Standard (from PCR) See 1 in Table Exceptions
Standard	Standard	Standard/ Inactive	Standard/Inactive	Standard/Inactive
Standard	Customized	Standard	Customized	Customized
Standard	Customized	Standard/ Inactive	Customized/ Inactive	Customized/ Inactive
Standard	Customized	Revised	Revised (conflict) See 2 in Table Exceptions	Customized
Not applicable	New	Not applicable	New	New
Not applicable	Not applicable	New	New	New
Standard	Deleted	Standard	Standard	Standard
Standard	Deleted	Standard/ Inactive	Standard/Inactive	Standard/Inactive
Standard	Deleted	Revised	Revised	Revised

### Table Exceptions

Exceptions listed in [Table 28](#) are as follows:

- In an ICL merge, UI navigation changes at Siebel 7.7 are implemented in screens copied-in from the Prior Customer Repository.
- A conflict exists when an object is different in all three repositories. The merge process typically resolves the conflict in favor of the New Siebel Repository. After the merge, you can review conflicts and change the resolution.

### Some UI Objects Are Not Eligible for ICL Handling

Certain screens, views, and applets that are required to administer applications and support the standard operation of the application are never eligible for ICL handling. These include UI objects for Server Administration and Remote Administration.

In addition, if the screens, views, and applets in an application have changed significantly in a new release, they are also excluded from ICL handling. ICL handling for these UI objects is not practical, because their layout and functionality have been greatly revised.

Application areas excluded from ICL handling in recent releases are as follows:

- **Siebel 7.7:** Applets and views for Siebel Employee Relationship Management (ERM) and Siebel Marketing.

### Upgrade Behavior Property

As of Siebel 7.7, an object property called Upgrade Behavior determines whether a UI object is preservable:

- If this property is set to NULL or Preserve then the UI object is eligible for ICL handling.
- If this property is set to ADMIN, then the UI object is not eligible for ICL handling.

The Upgrade Behavior property is defined on Screen, View, and Applet.

At each release, Oracle sets the value of Upgrade Behavior for UI objects. Do not change these values.

### How ICL Affects the Overall UI

In general, an ICL merge preserves the UI layouts of the Siebel Release you are upgrading from but also applies any significant UI navigation changes in the new Siebel Release.

#### Screens

The association of views with screens is preserved. For upgrades prior to Siebel 7.7, a new navigation scheme is introduced. When reviewing the UI after the merge, verify that all views can be displayed.

#### Views

- The association of applets with views is preserved.

- The layout of views is preserved.
- System pages such as Help, About View, About Record, and Technical Support are not preserved.

### Applets

The Web templates of standard and customer-modified applets in the Prior Customer Repository are preserved:

- When a Web template is used for an applet. Exception: for upgrades prior to Siebel 7.7, the postmerge utilities provide a new Web template for MVG applets that the postmerge utilities have converted to shuttle-enabled.
- All Web template item properties are preserved:
  - All control types are preserved.
  - The controls that are visible are preserved.
  - No new controls from the New Siebel Repository are added or deleted. Exception: for upgrades prior to Siebel 7.7, MVG applets receive new controls as part of being shuttle-enabled.
  - The location of controls in applets is preserved.
  - Applet mode (whether an applet displays in More mode or Less mode) is preserved.
  - Exception: for upgrades prior to Siebel 7.7, many flow-based form applets are converted to grid-based.
- Applet Drilldowns and their child objects are preserved.
  - Drilldown properties are preserved.
  - The columns that have drilldowns is preserved
  - The destination views of drilldowns are preserved

The following features are not preserved:

- Applet-level and application-level menus.
- Properties of controls or list columns. For example, row height, column width, caption, and pop-up icons are not preserved.
- As of Siebel 7.7, Pick and Association applets support in-line queries. This feature is implemented whether or not you choose ICL.

If you are upgrading from a Siebel 7.x release prior to Siebel 7.7, ICL has the following specific effects on applet controls and layout:

- Siebel 7.7 added two buttons in attachment applets: New File and New URL. An ICL merge does not add these buttons to applets.
- As of Siebel 7.7, MVG dialog boxes display only an OK button to close them. After an ICL merge, MVG dialog applets will have both OK and Cancel buttons.
- Siebel 7.7 added two new UI features, Query Assistant and Quick Print. An ICL merge does not add these buttons to applets. You must add these buttons as desired after completing the upgrade.

## How the Siebel Database Upgrade Works ■

- In Siebel 7.7, applets have three default buttons: New, Delete, and Query. If you select an ICL merge, applets will keep their existing default buttons.
- In Siebel 7.7, the Reset button was removed from applets. If you select an ICL merge, applets keep the Reset button.

### Navigation

Changes to navigation in the release you are upgrading to are implemented. This may affect screen, view, and applet layouts.

For example, in Siebel 7.7, a new navigation scheme was introduced. If you are upgrading from a release prior to Siebel 7.7, this navigation scheme is implemented during the upgrade regardless of whether you choose ICL. In parent list views, view tabs do not display in a row near the middle of the view as they did in 7.0x and 7.5x. You must select a record in the parent list applet to display the view tabs.

### How ICL Handles Deleted and Obsolete Objects

An ICL merge treats deleted UI objects in the same way as a regular merge. If you deleted screens, views, or applets in the Prior Customer Repository, and they are present in the New Siebel Repository they will be present in the New Customer Repository after the merge. You will need to manually verify and remove these as desired.

In a regular merge, if an object is obsolete (inactive) in the New Siebel Repository, the object is inactive in the New Customer Repository after the merge. This is also true for an ICL merge. In addition, an ICL merge replaces the following object types with their counterparts from the Prior Customer Repository. The replacement object types are then set to inactive:

- Applet Toggle
- Control and child objects
- Drilldown and child objects
- List and child objects
- Chart and child objects
- Tree and child objects

Any other UI object types that are obsolete in the New Customer Repository are handled the same way as a regular merge.

### ICL and Customer-Created UI Objects

If you created screens, view, or applets in the Prior Customer Repository, they are handled as follows:

- The ICL merge step ignores the Upgrade Ancestor property. For example, New Applet specifies Standard Applet as an upgrade ancestor. New Applet does not receive ICL handling. This means that New applet and all its child objects are copied intact to the New Customer Repository. The child objects do not replace UI objects in the New Customer Repository.

- For an ICL merge, the postmerge utilities copy new Web template files to the Tools installation directory. These files replace those in the new release and will be used to render customer-created objects.

## ICL and the Upgrade Ancestor Property

The ICL merge step ignores the Upgrade Ancestor Property. Customer-created applets do not receive ICL handling. For example, you create Applet-B by copying standard Applet-A. In Applet-B you specify Applet-A as the upgrade ancestor. In the New Siebel Repository, Applet-A has a new button. After an ICL merge, Applet-B displays the new button, but Applet-A does not:

- During the regular merge, Applet-B receives the new button, because it is a descendant of Applet-A.
- During the ICL merge portion, the Web template configuration of Applet-A is copied from the Prior Customer Repository to the New Customer Repository. This Web template configuration does not contain the new button. This removes the new button from Applet-A.
- Because the ICL merge portion ignores the Upgrade Ancestor property, this change is not propagated to Applet-B. Applet-B retains the new button.

In a regular merge, both Applet-A and Applet-B display the new button.

On an ICL merge, the regular merge portion propagates upgrade handling of non-ICL-object types (Business Component, and so on) to descendants normally. For this reason, using the Upgrade Ancestor Property with an ICL merge is recommended.

## How an ICL Merge Works

An ICL merge is a regular merge that contains two additional steps:

- The first step occurs at the end of the merge.
- The second step is part of the postmerge utilities, which run after the merge is complete.

### Additional Merge Step

At the end of the regular merge portion, the ICL merge step runs and makes changes to ICL objects in the New Customer Repository:

- ICL objects from the new release, are inactivated, and -UPG is appended to their name. These objects are read-only and cannot be deleted.
- ICL objects from the Prior Customer Repository are copied to the New Customer Repository and are verified as active.

The ICL merge step writes status and issues to the merge log files.

### Postmerge Utilities and Grid-Based Applets

The second step occurs after the ICL merge step completes and the merge is finished. The postmerge utilities run and make additional ICL changes:

## How the Siebel Database Upgrade Works ■

- At Siebel 7.7, many form applets in the New Siebel Repository that were formerly flow-based are grid-based. After an ICL merge, flow-based form applets from the Prior Customer Repository will replace some of these grid-based applets in the New Customer Repository, particularly for upgrades from releases prior to Siebel 7.7. The postmerge utilities will convert these replacement flow-based applets to grid-based.
- For flow-based form-applets that are converted to grid-based, you can choose to put field labels on top or on the left when setting up the merge. The UI standard as of Siebel 7.7 is to place field labels on the left.

### Postmerge Utilities and Web Template Files

The postmerge utilities copy-in new Web template files. These Web template files are similar to those in the release you are upgrading from. They provide page containers and supporting Web templates that preserve screen, view, and applet layouts. They also implement navigation changes. If you have edited the Web template files in the release you are upgrading from, you must reimplement your customizations in the new Web template files.

The postmerge utilities make the following changes to Web template files in the Siebel Tools installation directory:

- The utilities move the Web template files for the new release from `webtempl` to `\temp\webtempl`.
- The utilities copy Web template files from a subdirectory of `repatch\web_templates` to `\webtempl`.

These Web template files support the UI layout of the release you are upgrading from. They are similar but not identical to the Web template files included in that release. For example, if you selected the 7.5.3 and Label on Top ICL options, the postmerge utilities copy Web template files from the `753` and `TopLabel` subdirectories of `repatch`.

The postmerge utilities write status and issues to the postmerge utilities log.

### ICL and HTML Style Sheets

The postmerge utilities do not copy-in a new version of the HTML style sheets, `main.css` and `printmain.css`. The style sheets for the new release are used to render all UI objects, including those copied-in from the Prior Customer Repository. If you customized the style sheets in the release you are upgrading from, you must reimplement your customizations in the style sheets in the new release.

### What Happens at the Next Siebel Release?

You can perform an ICL merge for only every other upgrade. For example, if you performed an ICL merge when upgrading from Siebel 7.x to Siebel 7.7, performing an ICL merge when upgrading to the next release is not recommended.

At the next release before you run the repository merge, you must run a Siebel Tools utility that restores repository objects to the standard UI. The utility makes the following repository changes:

- Removes the ICL objects that were copied-in from the Prior Customer Repository.

- Activates the ICL objects with the -UPG suffix and removes the suffix. These –UPG objects are from the Prior Customer Repository (the repository in the release you are upgrading from). This reverses the effect of ICL and upgrades your UI to the current release.

A Tools utility provides a method for defining a filter to identify UI objects that you have modified since the prior upgrade. If an object has been modified since the prior upgrade, it is not replaced by its corresponding -UPG object. You must manually reconfigure these customer-modified objects after the merge as needed.

### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

[“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“Reviewing Siebel Repository Object Property Conflicts” on page 238](#)

## About the Siebel Database Server

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

The executables for doing an upgrade reside on the Siebel Server. You can use any upgraded Siebel Server to perform an upgrade of the Siebel database.

The Siebel Database Server is not a server. It does not participate in processing business data like a Web server or Siebel Server. Instead, the Siebel Database Server is a set of files and scripts that provide inputs to the upgrade executables.

For best performance, install the Siebel Database Server files on the Siebel Server that you will use to perform the upgrade. You must manually select the Siebel Database Server for installation; it is not automatically installed when you install a Siebel Server.

The Siebel Database Server files will be installed at the same directory level as the Siebel Server in a directory called dbsrvr. For example, if the Siebel Server is installed in C:\sea7xx\si ebsrvr (Windows), then the Siebel Database Server will be installed in C:\sea7xx\dbsrvr. You need to install only one Siebel Database Server.

To edit and execute Siebel Database Server procedures and maintenance scripts, you must have READ-WRITE access to the Siebel Server bin directories in *SIEBEL\_ROOT* (Windows), *\$SIEBEL\_ROOT* (UNIX). You must also have READ-WRITE access to the Siebel Server Log directories and upgrade directory.

## About Tuning Siebel Production Upgrade Files

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only.

**Databases:** All databases.

The Upgrade Tuner allows you to tune the upgrade files generated by the Siebel Database Server Configuration Utilities. Tuning the production upprep files can significantly reduce database downtime when performing a production environment upgrade.

Upgrade Tuner displays the following information and provides the following options:

- **Table creation times.** You can place tables with long creation times in parallel threads.
- **Index creation times.** You can assign index creation to run in parallel threads.
- **Zero-Row SQL commands.** You can review and inactivate SQL statements that do not affect any table rows.

The scripts used to upgrade your Siebel database are generic. They update your Siebel database to support all Siebel applications functionality. You can use this option to eliminate SQL statements that are not needed for your applications.

### Operating System and RDBMS Support

Upgrade Tuner supports parallel threads for table and index creation for the combinations of operating system and RDBMS show in [Table 29](#).

Table 29. Upgrade Tuner Support for Parallel Table and Index Threads

Operating System	Oracle	IBM DB2	Microsoft SQL
Windows	Yes	No	No
UNIX	Yes	No	No

Upgrade Tuner is not supported for IBM z/OS. Upgrade Tuner supports zero-row SQL command deactivation for the combinations of operating system and RDBMS shown in [Table 30](#).

Table 30. Upgrade Tuner Support for Zero-Row SQL Deactivation

Operating System	Oracle	IBM DB2	Microsoft SQL
Windows	Yes	Yes	Yes
UNIX	Yes	Yes	Yes
z/OS	Not applicable	No	Not applicable

For information on which versions and releases of the operating systems and RDBMS products that Oracle supports for Siebel products, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

Upgrade Tuner is part of the Siebel Server and runs only under Microsoft Windows. Upgrade Tuner does not run under UNIX. To tune UNIX production upgrade files, you must copy them to a Windows platform, tune them, and move them back to the UNIX platform. Scripts are provided to move the files.

If you are a UNIX user and do not have a Siebel Server for Windows, contact your account manager or Oracle to obtain one.

### When to Use Upgrade Tuner

Use Upgrade Tuner in the production test environment to tune the upgrade files that perform the production upgrep. There is no need to tune the upgrade files that perform the production upgphys. You also do not need to tune the upgrade files that perform the development environment upgrep or upgphys.

### Upgrade Tuner Modes

When you start Upgrade Tuner it displays four tabs:

- Process Information
- Parallelize Table Creation
- Parallelize Index Creation
- Deactivate 0-Row SQLs

#### Process Information Tab

This page displays the information sources that Upgrade Tuner is using. These sources include the Logparse summary.xml file, master .ucf file, and driver .ucf file.

You cannot edit the information on this page. Upgrade Tuner obtains the information by reading the summary.xml file.

#### Parallelize Table Creation and Parallelize Index Creation Tabs

The Parallelize Table Creation page and the Parallelize Index Creation page both have the same layout. These pages allow you to do the following:

- **Parallelize Table Creation tab.** This page displays the time required to create tables and allows you to assign table creation to parallel threads. Adding a table to a parallel thread does not add index creation for that table to the thread. Table and index creation are handled as separate steps during the upgrade.
- **Parallelize Index Creation tab.** This page displays the time required to create table indexes and allows you to assign index creation to parallel threads.

Creating parallel threads improves the upgrade performance by reducing the total time to create tables and indexes. You can create up to nine parallel threads. Each thread can have a maximum of ten tables or indexes. Tables or indexes not assigned to a parallel thread remain in the serial thread.

### Deactivate 0-Row SQLs

This page allows you to activate or deactivate the SQL statements that do not affect any table rows and therefore any data. This capability improves the upgrade performance by eliminating SQL statements that may not apply to your data. This page lists only the SQL files that are executed natively by the RDBMS. It does not list SQL files that are executed using odbcsql.

### Files Required to Run Upgrade Tuner

Upgrade Tuner requires the following files. The location of the files is the same on both Windows and UNIX hosts.

File	Location on a Windows Host
summary.xml	<i>SIEBEL_ROOT\log\upgprep_prod_VERSION</i> For example, <i>SIEBEL_ROOT\log\upgprep_prod_77\summary.xml</i>
master_upgprep_prod_VERSION.ucf	<i>SIEBEL_ROOT\bin\</i> For example, <i>SIEBEL_ROOT\bin\master_upgprep_prod_77.ucf</i>
schema*.ddl	<i>DBSRVR_ROOT\DBPLATFORM</i> For example, <i>DBSRVR_ROOT\oracl e\schema.ddl</i> , <i>schema_i1.ddl</i> , <i>schema_t1.ddl</i>
driver_upgprep_prod_VERSION.ucf	<i>DBSRVR_ROOT\DBPLATFORM\upgrade\VERSION</i> For example, <i>DBSRVR_ROOT\oracl e\upgrade\77_7\driver_upgprep_prod_77.ucf</i>
*.sql	<i>DBSRVR_ROOT\DBPLATFORM\upgrade\VERSION</i> For example, <i>DBSRVR_ROOT\oracl e\upgrade\77_7\pret.sql</i> , <i>preschm.sql</i>

### How Upgrade Tuner Modifies Files

When you save your changes, Upgrade Tuner modifies the upgrade files. These are the files the Upgrade Wizard uses to upgrade the database. These files are as follows:

- [“Driver Configuration File” on page 112](#)
- [“Schema.ddl File” on page 113](#)
- [“SQL Files” on page 114](#)

#### Driver Configuration File

When you add or remove parallel threads and save your changes, Upgrade Tuner modifies the driver configuration file, for example *driver\_upgprep\_77.ucf*. The driver configuration file is a text file that contains a series of steps. The steps specify the commands that control the production upprep. When you run the Upgrade Wizard to upgrade your database, it executes the steps in the driver configuration file.

The following actions are examples of steps that can appear in the driver configuration file:

- Making schema changes using the `ddlmp` utility and `schema.ddl`
- Making schema and data changes by executing SQL scripts

Upgrade Tuner manages the driver configuration file as follows:

- When you save your changes after the first session, Upgrade Tuner makes a copy of the file and appends `.orig` to the file name. It then modifies the file. For example, Upgrade Tuner copies `dri ver_upgprep_77.ucf` to `dri ver_upgprep_77.ucf.ori g`. It then makes changes to `dri ver_upgprep_77.ucf`.
- When you save your changes after the second session, Upgrade Tuner makes a copy of the file and appends `.old` to the file name. It then modifies the driver file. For example, Upgrade Tuner copies `dri ver_upgprep_77.ucf` to `dri ver_upgprep_77.ucf.ol d`. It then makes changes to `dri ver_upgprep_77.ucf`.
- When you save your changes after the third session and all following sessions, Upgrade Tuner saves the driver file to `.old` again and then updates the driver file.

This file management strategy preserves the previous set of revisions to the file. It also preserves the original version of the file.

### Schema.ddl File

When you run the Upgrade Wizard after the production `upgprep` it reads the driver file. The driver file contains steps that call the `ddlmp` utility. This utility uses `schema.ddl` as input to upgrade your database schema.

In the Parallelize Table Creation and Parallelize Index Creation pages, Upgrade Tuner displays the creation times for the tables and indexes in the `schema.ddl` file.

When you create parallel threads, Upgrade Tuner creates thread-files that have the same format as `schema.ddl` but contain only the table or index creation steps in the thread. Upgrade Tuner then adds steps to the driver file. These steps call the `ddlmp` utility, and specify the thread-files as input.

Upgrade Tuner manages `schema.ddl` and thread-files as follows:

- When you create a new thread and click `Save and Exit`, Upgrade Tuner creates a `schema.ddl` thread-file for the new thread.

For example, you do not have any parallel threads, and then create two new threads for table creation. When you exit, Upgrade Tuner creates a `schema_t1.ddl` and a `schema_t2.ddl` file. Upgrade Tuner also inserts steps in the driver file to execute the thread-files.

The `t1` thread-file contains the table creation information for the tables in Parallel Thread 1. Parallel thread 2 information is contained in the `t2` thread-file, and so on.

- When you create new threads in the Parallelize Index Creation page, the thread-files are named `i1`, `i2`, and so on. For example, the information for Parallel Thread 1 for index creation is contained in `schema_i 1.ddl`.

## How the Siebel Database Upgrade Works ■

- If you run Upgrade Tuner and change the tables or indexes assigned to a thread, Upgrade Tuner updates the thread-file for that thread. Upgrade Tuner does not create .orig or .old files for thread-files. Also, Upgrade Tuner does not change the step that executes the thread-file in the driver file.
- Upgrade Tuner does not revise the content of the main schema.ddl file (the serial thread) when you create thread-files. The thread-files duplicate the content in schema.ddl.

In the driver file, the order of execution of steps for schema.ddl and the thread-files is as follows:

- Table thread files beginning with file t1 (ddlmp in table creation mode)
- schema.ddl (ddlmp in table creation mode)
- Index thread files beginning with i1 (ddlmp in index creation mode)
- schema.ddl in index creation mode (ddlmp in index creation mode)

For both table and index creation, the parallel threads are executed first followed by the serial thread (schema.ddl).

**TIP:** To locate thread-file steps in the driver\_upgprep\_prod ucf file, query for "schema\_".

### SQL Files

When you make changes in the Deactivate 0-Row SQLs page, Upgrade Tuner makes changes to the SQL file containing the SQL command. Because the SQL file is already a step in the driver file, Upgrade Tuner does not modify the driver file.

Upgrade Tuner manages the SQL files as follows:

- When you first change an SQL file, Upgrade Tuner saves a copy of the SQL file and appends .orig to its file name. Upgrade Tuner then updates the SQL file.
- The next time you change the SQL file in Upgrade Tuner, it saves a copy of the SQL file and appends .old to the file name. Upgrade Tuner then updates the SQL file.
- Thereafter, when you modify the SQL file, Upgrade Tuner saves the SQL file to .old again and then updates the SQL file.

This file management strategy preserves the previous set of revisions to the file. It also preserves the original version of the file.

# 5

## Siebel Database and UI Upgrade Planning

This chapter contains the following topics:

- [“About Siebel Upgrade Planning Resources”](#) on page 115
- [“About Best Practices for Doing Your Siebel Database Upgrade”](#) on page 118
- [“About New Siebel Upgrade Features and Requirements”](#) on page 121
- [“About Upgrading Your RDBMS in the Siebel Environment”](#) on page 121
- [“About Siebel User Interface Changes”](#) on page 122
- [“About Database Sort Order in the Siebel Environment”](#) on page 123
- [“About the Siebel Party Model”](#) on page 124
- [“About Migrating Siebel HTML Attachments to Base Tables”](#) on page 127
- [“About Siebel Database Growth Estimates”](#) on page 128
- [“Upgrade Planning for Multilingual Siebel Deployments”](#) on page 129
- [“Upgrade Planning for Siebel Unicode Support”](#) on page 130
- [“Upgrade Planning for Siebel RC2 or AES Encryption”](#) on page 132
- [“Upgrade Planning for Siebel Web Template Files and Style Sheet”](#) on page 132
- [“Upgrade Planning for Siebel Access Control”](#) on page 134
- [“Upgrade Planning for Siebel Database Clustered Indexes”](#) on page 135

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## About Siebel Upgrade Planning Resources

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic lists important publications and resources for performing an upgrade. Review these as part of the upgrade planning process.

## Product Documentation

Oracle's Siebel Product documentation is collectively called the Siebel Bookshelf. The Siebel Bookshelf is available on CD-ROM. It is also available on Siebel SupportWeb under Product Documentation.

## Siebel Installation and Deployment Documentation

The following publications are meant to be used in concert with installation or deployment of the Siebel application and can be found on *Siebel Bookshelf*.

- *Deployment Planning Guide*
- *Siebel Installation Guide for Microsoft Windows: Servers, Mobile Web Clients, Tools*
- *Siebel Installation Guide for UNIX: Servers, Mobile Web Clients, Tools*
- *Going Live with Siebel Business Applications*. Provides information about how to migrate customizations from the development environment to the production environment
- *Siebel System Administration Guide*. Details on how to administer, maintain, and expand your Siebel Servers
- *Security Guide for Siebel Business Applications*
- 
- *Performance Tuning Guide*
- *Configuring Siebel Business Applications*. Provides information about configuring Siebel Business Applications in Siebel Tools
- *Siebel Data Model Reference (DMR)*. The DMR describes in detail the Siebel database schema for a Release. It also lists certain types of schema changes. Use the DMR during upgrade planning to evaluate how data will be stored in the new Release. Consider obtaining a DMR for both the Release you are upgrading from as well as the Release you are upgrading to. Contact your account representative or Siebel Technical Support for information on ordering DMRs.
- **6.x upgrades only:** *Planning an Upgrade to Siebel 7* for information about upgrading to the Siebel 7 data model for the first time
- **6.x upgrades only:** *Siebel Connector for SAP/R3* to obtain Siebel 6.x upgrade instructions specific to this connector

## Siebel SupportWeb

This is Oracle's Siebel products technical support Web site. It provides search engine access to the Siebel Bookshelf, Technical Notes, Alerts, troubleshooting information and other important information. SupportWeb is located at <https://ebusiness.siebel.com/supportweb/>.

- *Installation and Upgrade Portal Page*. This SupportWeb page lists important installation and upgrade publications, including Bookshelf titles, Technical Notes, and Alerts. Use this page as your entry point for obtaining installation and upgrade information. It is located on Siebel SupportWeb > Installation and Upgrade.

- *System Requirements and Supported Platforms* on Siebel SupportWeb *System Requirements and Supported Platforms* on Siebel SupportWeb. This document is the definitive list of system requirements and supported third-party products. It is located on Siebel SupportWeb at Product Documentation > System Requirements and Supported Platforms & Miscellaneous Documentation.
- *Release Notes*. *Release Notes* contain late-breaking information that the *Siebel Database Upgrade Guide* does not yet include. Release Notes regarding upgrade are located on Siebel SupportWeb at Product Documentation > Release Notes > Siebel Business, Industry, Midmarket Edition, and CRM Applications 7.x and later > Categories > General > Upgrade.
- *Maintenance Release Guides*. Maintenance Release Guides contain important information about updates to applications in maintenance releases. Maintenance Release Guides are located on Siebel SupportWeb at Product Documentation > Maintenance Release Guides.
- *Documentation Updates*. Typically, the Bookshelf is updated monthly. During the month, PDF updates are posted to Siebel SupportWeb > Product Documentation > Documentation Updates.
- *Technical Notes and Alerts*. Technical notes and Alerts provide important information on specific upgrade issues. Technical Notes related to upgrade are located on Siebel SupportWeb at Technical Notes > Upgrade. Alerts related to upgrade are located on Siebel SupportWeb at Alerts > Upgrade.  
  
Alert 1002 and Alert 1179 are master Alerts that list Technical Notes and Alerts particularly important to all upgrades. References to these Alerts and Technical Notes are integrated throughout the *Siebel Database Upgrade Guide*.
- *Troubleshooting Steps*. Troubleshooting Steps contain information about how to troubleshoot common error messages and unwanted behavior in Oracle's Siebel applications. Troubleshooting Steps are for upgrade list error messages found in upgrade logs and describe how to resolve them. Troubleshooting Steps are located on Siebel SupportWeb at Troubleshooting Steps > Product Areas > Upgrade.
- *Siebel Weekly Content Notification Service*. This service notifies you weekly by email of important content changes on Siebel SupportWeb. This includes new product documentation, technical notes, alerts, and troubleshooting steps. To subscribe, see the Siebel Weekly Content Notification banner on the SupportWeb search page.

### Technical Account Manager

If you need assistance planning your upgrade or encounter problems during the upgrade, your Technical Account Manager can advise you on how best to use available Oracle resources.

### Expert Services

Oracle's Siebel Expert Services offers detailed implementation planning and technical consulting services. They also provide rapid response and resolution for critical technical issues affecting Siebel deployments.

Contact Oracle's Siebel Expert Services for important information in the following areas:

- Migrating from Siebel Industry Solutions or Siebel Financial Services to Siebel Business Applications during upgrade

- Migrating to Unicode code page support during upgrade
- Changing operating system type during upgrade

## About Best Practices for Doing Your Siebel Database Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic provides an overview of the best practices for planning upgrade resources, estimating the upgrade time line, and managing the data migration process.

If you are upgrading from Siebel 6.x, see *Planning an Upgrade to Siebel 7* for upgrade considerations and best practices for upgrades to the Siebel 7.x data model.

Use the following steps to help plan your upgrade.

- 1 Determine your upgrade path.** First, refer to *System Requirements and Supported Platforms* on Siebel SupportWeb to determine supported upgrade paths for major releases. Verify whether you can upgrade directly to the currently shipping release or whether you must upgrade to a previous release first. Second, refer to the Maintenance Release Guides on Siebel SupportWeb for the release to which you are upgrading. These Guides list the upgrade path from the major release to its most recent maintenance release. Maintenance releases roll up Fix Packs and also include new features. Because fixes to the upgrade process may exist in Fix Packs included in a maintenance release, you should install the most recent maintenance release before starting the upgrade.
- 2 Evaluate the complexity of the upgrade.** Determine the complexity of the upgrade effort based on Oracle's Siebel modules implemented, number of integration points, number of interfaces, total number of scripts, and number of user interface scripts.
- 3 Assess the current Siebel environment and evaluate the existing implementation.** Perform a detailed assessment of the current Siebel environment to determine how the implementation will be affected by the upgrade. Evaluate the current implementation in comparison with the architecture of the current release. The assessment will help you to identify areas where you can take advantage of new functionality to meet business requirements.
- 4 Estimate the level of effort to upgrade.** Determine the metrics and cost associated with each aspect of the upgrade. Determine the effort required to upgrade based on the results of your complexity evaluation, current environment assessment, and new functionality review. This will help you to estimate resources, time line, and costs.
- 5 Establish the upgrade team.** Assemble a cross-functional upgrade team that understands Siebel product architecture and performance best practices. Include IT professionals, executives, and users to ensure a broad base of experience in technical, business, and Siebel-specific skills.
- 6 Review interface migration tasks.** Determine the effort to migrate modified applets and views. This includes associating applets with Web template items and mapping them to Web template controls.

- 7 **Plan for upgrade tuning.** Tuning your production upgrade scripts can significantly reduce downtime during the final stages of your upgrade. Examples of upgrade tuning include eliminating SQL statements that do not affect any data, executing long-running SQL statements in parallel, and executing table creation, table rebuilds, and index creation in parallel.
- 8 **Identify data migration tasks.** After the upgrade, there may be data migration and repository configuration tasks that must be performed manually. These tasks frequently involve customizations made in prior releases.
- 9 **Plan for end-user training.** Analyze the impact of change on the users, and develop a plan for end-user training and adoption.

The upgrade of your application requires several key things to be successful:

- A detailed understanding of customizations made to your current deployment
- Analysis and definition of the components within your enterprise
- Analysis of how to use new functionality provided by Oracle's Siebel software
- Strict adherence to industry best practices and best practices identified in this guide

The upgrade planning process will produce a road map for the entire upgrade project that outlines infrastructure, deployment, and training requirements.

Use the results of this process to develop a project plan that identifies required skills and resources for developing and deploying the upgraded application. This will help you with advance budgeting of resources, time, and training.

## Upgrade Planning Best Practices

Here are important best practices to follow when planning an upgrade:

- Review *System Requirements and Supported Platforms* on Siebel SupportWeb, Release Notes, Maintenance Release Guide, and Alerts related to upgrades to verify your upgrade path. These documents are available on Siebel SupportWeb at <https://ebusiness.siebel.com/supportweb/>.
- Gather all relevant documentation that describes the current implementation, for example, requirements documents, design documents, and architecture context diagrams.
- Implement a change management program. For example, communicate rollout dates to users, schedule training, allow adequate time for users to adjust to the enhancements, and provide a process for end users to provide feedback to the project team.
- User adoption is critical to a successful upgrade. Provide access to a test environment that allows users to become familiar with the new version of the application, and provide end-user training on the upgraded application.
- Do not start a new development effort until after the new version has been rolled out. This is especially important if you are upgrading from Siebel 6.x to Siebel 7.7. There are significant user interface differences between the two releases, so you must change existing functionality before moving on to customized functionality.

## Database Planning Best Practices

Here are important best practices to follow when planning the upgrade of your database:

- Analyze the impact of the upgrade on table customizations that you have made. Determine if pre-upgrade data migration is required. Determine what postupgrade schema changes are required. For a list of tables that are affected by the upgrade, see [Appendix B, “Tables Modified or Seeded During a Siebel Upgrade.”](#)
- Consider database layout in your planning. Plan to tune the database and database server for the upgrade, because settings and parameters for upgrade differ from those required for OLTP.
- If you are migrating multiple languages from a prior version, plan extra time (one to two weeks) for the repository merge process. The expected merge time may increase with the number of languages in the repository. You also might need to plan for additional installation-related tasks.
- For IBM DB2 platforms, consider increasing the size of your tablespaces before going live. Make sure that your custom tablespaces are large enough for upgraded tables. See [“Analyzing IBM DB2 Custom Tablespace Requirements for a Siebel Upgrade” on page 162.](#)

## Production Database Upgrade Best Practices

Both the Prepare for Production step and the upgrade tuning process modify the SQL scripts used to perform the upgrade in the production test environment. In addition, it is common to further modify these scripts to meet local requirements.

The recommended way to perform the production upgrade is to use the SQL scripts that you have generated and modified for the production test upgrade. The steps in the upgrade process checklist for the production upgrade implement this as follows:

- Run the Database Server Configuration Utilities in the production test environment.
- In the Utilities, enter the information for the production environment instead of the production test environment. For example, you enter the ODBC connection for the production environment.  
  
This configures the driver file to run against the production database rather than the production test database. It also configures the driver file to use the upgrade SQL files you generated for the production test upgrade.
- Run the Upgrade Wizard. The Upgrade Wizard uses the SQL files in the production test environment to upgrade the database in the production environment.

This approach has several advantages:

- You do not have to generate upgrade SQL files in the production environment and then manually transfer customizations to them from the production test environment.
- You do not lose any changes to the SQL files that were made by Siebel Upgrade Tuner in the production test environment.
- You do not have to run the Database Server Configuration Utilities in Prepare for Production mode again.
- With some exceptions, you do not have to perform database-related configuration tasks required by Release Notes or Alerts again.

If your network configuration prevents creating an ODBC connection to your production database from inside your production test environment, contact Siebel Technical Support for assistance in completing the production upgrade.

## About New Siebel Upgrade Features and Requirements

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.



The following new upgrade features are introduced at Siebel 7.7:

- **Incorporate Custom Layouts (ICL).** This feature enables you to preserve the UI look and feel of your currently installed release during the repository merge, reducing the workload required to reconfigure customizations after the upgrade. ICL is available for Siebel 7.x upgrades only; however, it is included in Siebel 7.7.2 and later. For more information, see [“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#).
- **Log Parsing Utility.** The log parsing utility analyzes many of the logs generated during the upgrade and provides a report that lists problems. For more information, see [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219](#).
- **Production Upgrade Tuner.** This utility analyzes the SQL commands used to upgrade the production test environment. It then helps you eliminate unneeded SQL commands and perform other operations intended to improve the performance of the production environment upgrade. For more information, see [“About Tuning Siebel Production Upgrade Files” on page 109](#).
- **Upgrading the Production Database.** In the production test environment, you must be able to define an ODBC connection to the Siebel Database in the production environment to do the production upgrade. If you cannot do this, contact Oracle’s Siebel Technical Support. For more information, see [“About the Siebel Database Upgrade Process” on page 68](#).

## About Upgrading Your RDBMS in the Siebel Environment

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

If your currently installed RDBMS version is not supported in the new release, you must upgrade your RDBMS before performing the Siebel database upgrade. For information on supported RDBMS versions, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

If your currently installed RDBMS version is supported in the new release, and you plan to upgrade your RDBMS, you can do so before or after upgrading your Siebel Database.

The following guidelines and requirements are for planning your RDBMS upgrade.

**NOTE:** When you upgrade the RDBMS, also be sure to upgrade your client database connectivity software. See *System Requirements and Supported Platforms* on Siebel SupportWeb.

### Oracle

- See Alert 1053 on Siebel SupportWeb for planning information on upgrading an Oracle RDBMS. If you upgrade the Siebel Database and then upgrade the Oracle RDBMS, you must validate the Siebel Database schema after the upgrade. See [“Verifying an Upgraded Oracle RDBMS After a Siebel Upgrade” on page 309](#).
- If you plan to upgrade your RDBMS to Oracle 10g as part of upgrading to Siebel 7.7.2, see Technical Note 640 for instructions on how to run the Siebel Software Configuration utility. For a full discussion on how to install a Siebel Release, see *Siebel Installation Guide* for the operating system you are using.
- Upgrades on Oracle versions 8i and 9i must be run in Rule-Based Optimizer (RBO) mode. If you are running in Cost-Based Optimizer (CBO) mode, you must change to RBO mode before upgrading. For Oracle 9i, you must return to CBO mode after the upgrade.
- Upgrades on Oracle 10g must be run in CBO mode.
- Oracle’s Siebel applications support Oracle 8i databases for Western European and Japanese languages only. For all other languages (such as Korean), Siebel applications support a Unicode character set for Oracle 9i.

If you are using an Oracle 8i database with a language other than Western European or Japanese, you must upgrade your Oracle database software to Oracle 9i before performing the Siebel Database upgrade. After the Siebel Database upgrade, contact Oracle’s Siebel Technical Support or Oracle’s Siebel Expert Services to migrate from a non-Unicode code page (or character set) to Unicode.

### IBM DB2

- After upgrading your IBM DB2 RDBMS, you must upgrade the database instance.
- The DB2 database must have 4-KB, 16-KB, and 32-KB tablespaces defined on it. Otherwise, your upgrade will not complete successfully.
- If you are upgrading from Siebel 6.x, you must install a new set of stored procedures and user-defined functions on the DB2 RDBMS server.

## About Siebel User Interface Changes

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Several important changes to the user interface were introduced in Siebel 7.7:

- **New navigation scheme.** The key revisions are as follows:
  - The Show menu is replaced by a link bar. When you click a screen tab, the link bar displays below the screen tab and shows the views or view list for the screen.
  - A visibility filter menu, located in the parent list applet, provides alternative methods for filtering the data displayed in the applet.
  - To display view tabs, users click on a link in the parent list applet.
- **Revised UI configuration.** In prior releases, the placement of views and controls was based on the repository configuration and logic applied at run time. In Siebel 7.7, the placement of views and controls is declarative in the repository. No run-time logic is applied.
- **Expanded use of grid-layout applets.** To improve usability, most employee-specific form applets are converted to grid-based layout. The upgrade does not convert custom form-based applets to grid-based layout.
- **MVG shuttle applets.** MVG shuttle applets were introduced in Siebel 7.5x. In Siebel 7.7, MVGs are shuttle-enabled by default.

Because of these enhancements, plan a careful evaluation of the user interface after the upgrade. The upgrade logging utility lists any user interface-related problems encountered during the upgrade. For more information on the user interface, see *Fundamentals*.

## About Database Sort Order in the Siebel Environment

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

Sort order (also called collation sequence) is specified during the initial installation of a database and defines the way in which the database sorts character data when executing queries. Sort order support depends on both the code page of the database and whether it is used in a development or a production environment:

- **Development environments.** For development databases, you must use a binary sort order due to the functional limitations of databases that use a nonbinary sort order.
- **Production environments.** For production databases, it is strongly recommended that you use binary sort order to prevent possible performance degradation.

The settings for binary sort order are unique for each database platform. See *System Requirements and Supported Platforms* on Siebel SupportWeb to determine supported settings for your database platform.

## Sort Order Considerations for Siebel Databases

If your deployment requires that you use a nonbinary sort order (for example, if your local language does not use binary sort order), you must consider several functional limitations that particularly affect development environment upgrades. If these limitations are unacceptable, consider re-creating your database to use binary sort order. However, be aware of these considerations:

- You cannot use Siebel Tools to generate a Siebel Repository file (SRF) on a database that uses a nonbinary sort order.
- You cannot perform a repository merge on a database that uses a nonbinary sort order.
- Databases that use nonbinary sort order might perform slower than databases that use binary sort order.

See *System Requirements and Supported Platforms* on Siebel SupportWeb to verify sort order and code page requirements for your deployment.

## About the Siebel Party Model

**Upgrades:** Applies to Siebel Financial Services upgrades from Siebel 6.x. Applies to Siebel Financial Services upgrades from Siebel 7.x that have retained the Siebel 6.x form of household associations.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Siebel 7.x introduced a party table (S\_PARTY) in which all persons and organizational units are held. Accounts, organizations, internal divisions, contacts, employees, positions, and households are all considered parties and can be referenced from this table.

Most of the tables that formerly contained this data still exist and are still used, but they are now extension tables to the S\_PARTY base table. Data is loaded into the business components through an implicit join.

Additionally, Siebel 7.x 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 same table (S\_ORG\_EXT).

The S\_PARTY table is the primary table in the Party or Single-Person model and is the base table for all Party business components.

Several extension tables support the Party Model:

- S\_USER stores Siebel User information.
- S\_EMP\_PER stores attributes for Brand-Owner Employees and Partner Users who are considered agents of the Brand-Owner.
- S\_BU stores Organization information.
- S\_CONTACT
- S\_ORG\_EXT

■ S\_POSTN

Each nonperson party directly or indirectly has person members, such as employees or contacts.

The Party model makes several tables obsolete:

- S\_EMPLOYEE. Its functionality is merged into S\_CONTACT.
- S\_ORG\_INT. Its functionality is merged into S\_ORG\_EXT.
- S\_EMP\_POSTN has been replaced by S\_PARTY\_PER.
- S\_EMPLOYEE\_ATT
- S\_ORG\_INT\_ATT
- S\_POSTN\_RPT\_REL

Figure 3 depicts the Party changes to the data model that occur during upgrades from Siebel 6.x to Siebel 7.x.

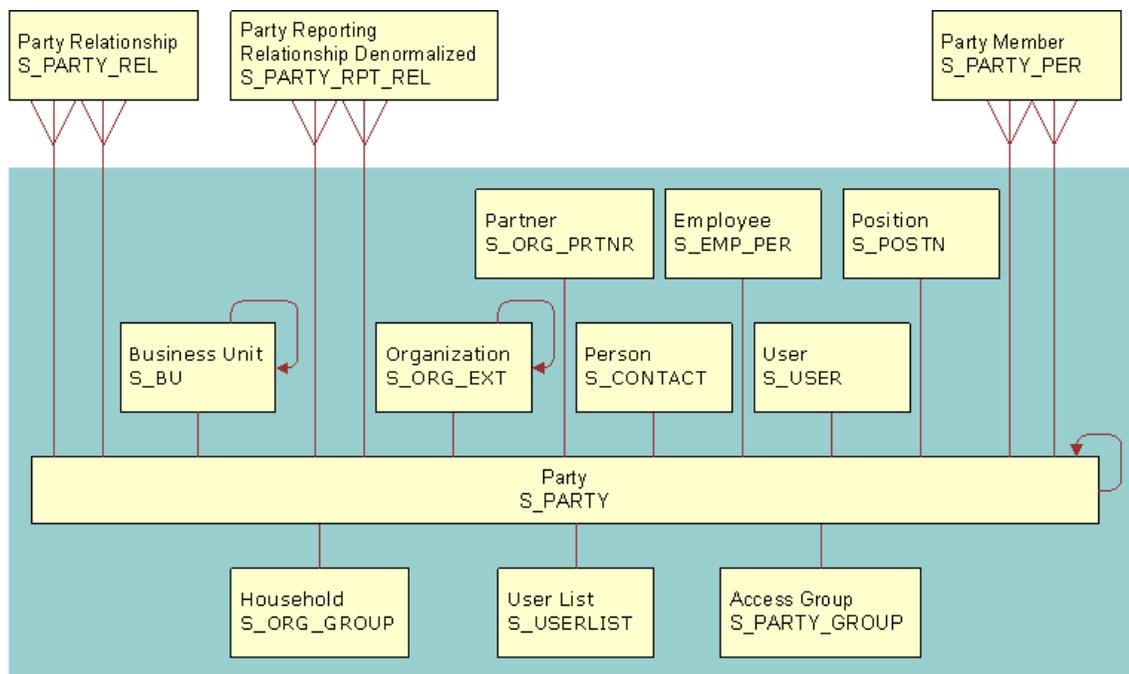


Figure 3. Party Model Changes

## How the Party Model Is Implemented During Upgrade

When you upgrade to Siebel 7.x, the upgrade process implements the Party model as follows:

### Data Migration

- Migrates data from S\_EMPLOYEE to S\_CONTACT, S\_USER, S\_EMP\_PER for standard columns
- Migrates data from S\_ORG\_INT to S\_ORG\_EXT, S\_BU for standard columns

- Creates S\_PARTY records for each previous contact, position, employee, account, division

### Business Component Definitions

- Updates business component definitions to reference S\_PARTY as the Primary Table (for example, Employee, Contact, Position, and Account business components)
- Changes standard and custom joins on S\_EMPLOYEE to S\_CONTACT, S\_USER, S\_EMP\_PER
- Changes standard and custom joins on S\_ORG\_INT to S\_ORG\_EXT
- Sets implicit joins 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 initiates the following actions:
  - Retargets Contact business component to S\_PARTY
  - Defines join to S\_CONTACT from S\_PARTY on Contact business component
  - Sets implicit join for the Alternate Phone field

### How the Party Model Affects Siebel Financial Services Household Data

For Siebel 7.0.x, the Party model changed the relationship between a household and the following entities for Siebel Financial Services:

- Policy/Financial Account
- Activity
- Claim
- Service Request
- Opportunity
- Company

The relationship changed as follows:

- In Siebel Financial Services 6.x, these entities could be associated directly with a household.
- As of Siebel Financial Services 7.0.x, these entities cannot be associated directly with a household. Instead, they are associated with a contact. You associate an entity with a household, by adding a contact associated with the entity to the household.

To implement direct relationships between the entity tables and a household table in Siebel Financial Services 6.x, intersection tables were used for many-to-many relationships. A foreign key was used for one-to-many relationships. This design allowed a contact to be assigned to an entity but not be part of the household assigned to that entity. This caused possible data integrity problems, which the Party model resolves.

The tables required for maintaining the 6.x direct-relationship design are retained in Siebel 7.x. You can choose to maintain direct relationships between households and entities. However, this is not recommended. To maintain the 6.x direct-relationship design, contact Siebel Technical Support or Oracle's Siebel Expert Services for assistance in revising business components in the current release to support this model.

## How Siebel Financial Services 6.x Household Data Is Migrated

During the upgrade to Siebel 7.x, relationships in S\_CONTACT\_REL are migrated to S\_PARTY\_REL. Relationships in S\_PER\_ORG\_UNIT are migrated to S\_PARTY\_PER.

When defining relationships between entities, no records are written to S\_PARTY. Instead, the "PARTY (FIN)" and "Party Relationship To" business components drive the relationships displayed in the Relationship Hierarchy applet and adjacent Party Relationship list applet:

- S\_PARTY includes the following fields:
  - PAR\_PARTY\_ID. This identifies the parent entity.
  - ROW\_ID
  - NAME
  - PARTY\_TYPE\_CD. This identifies the type of entity.
- S\_PARTY\_REL includes three fields:
  - PARTY\_ID. This identifies the entity that has the relationship (explicit owner).
  - REL\_PARTY\_ID. This identifies the entity that the relationship is with (implicit owner).
  - REL\_TYPE\_CD. This identifies the type of relationship.

S\_PARTY\_REL can be used to define custom relationships such as lawyer, accountant, board member, and influencer. Valid relationships can be created between the following entities:

- Contact-to-contact
- Contact-to-household
- Contact-to-company
- Company-to-contact
- Company-to-household
- Household-to-contact
- Household-to-household
- Household-to-company
- Company-to-company

## About Migrating Siebel HTML Attachments to Base Tables

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

The upgrade to Siebel 7.7 migrates data from the S\_WEB\_CNTNT table to the S\_CB\_CNTNT\_SET, S\_CB\_ASSET, and other Content Base tables.

HTML Attachments, introduced in Siebel 7, are upgraded into Content Base tables. The Content Base table system allows content to be stored in the database or in the Siebel File System. The system stores several types of content including plain text, markup languages (for example, SGML and XML), and image files (for example, GIF and JPEG). The system breaks up the content when storing it and reassembles the content before rendering it.

## About Siebel Database Growth Estimates

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

Database upgrade is resource intensive. If the upgrade exceeds available resources, the upgrade halts. You must then resolve resource issues before resuming the upgrade.

For estimates of growth percentages for specific RDBMS platforms and releases, see the following tables:

- IBM DB2: [Table 31 on page 128](#). Tablespace growth estimates include both table and index growth.
- Oracle: [Table 32 on page 129](#)
- MS SQL Server: [Table 33 on page 129](#)

Actual expected growth may vary widely from these estimates, depending on which Siebel application you are using (Siebel Business Application or Siebel Industry Application), database configuration, row size of tables, and data content.

Table 31. IBM DB2 Database Growth Estimates

If Upgrading from Release...	4-KB Tablespace	16-KB Tablespace	32-KB Tablespace
7.5x Unicode	15%-30%	230%-250%	130%-150%
7.5x	70%-80%	575%-600%	1100%-1200%
7.0x	35%-90%	25%-100%	30%-40%
6.3x	200%	400%	No change
6.2x	200%	400%	No change
6.0x	200%	400%	N/A

Table 32. Oracle Database Growth Estimates

If Upgrading from Release...	Tables	Indexes
7.5x Unicode	35%-50%	45%-60%
7.5x	40%-55%	10%-25%
7.0x	35%-80%	25%-55%
6.3x	40%-70%	70%-90%
6.2x	40%-70%	70%-90%
6.0x	40%-70%	70%-90%

Table 33. MS SQL Server Growth Estimates

If Upgrading from Release...	Tables	Indexes
7.5x Unicode	65%-80%	30%-45%
7.5x	40%-60%	60%-90%
7.0x	40%-90%	40%-100%
6.3x	N/A	N/A
6.2x	160%-175%	115%-130%
6.0x	160%-175%	115%-130%

## Upgrade Planning for Multilingual Siebel Deployments

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

*Global Deployment Guide*The upgrep process imports repository strings and seed data in only the primary (base) language. The primary language is the language in which data is stored in the Siebel Database.

If you have multilingual deployments, you must import multilingual repository strings and seed data after performing the upgrep. This applies to development, production test, and production upgrades.

See Technical Note 447 on Oracle's Siebel SupportWeb for instructions on how to perform these imports.

See the *Siebel Installation Guide* for the operating system you are using for instructions on installing and configuring multiple language packs.

For a list of supported code pages and encoding strategies, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

## Unicode Databases

In a Unicode-enabled database environment, you can install any available language pack. In a non-Unicode database environment, you must consider the correlation of the language packs you want to install and the characters supported by your RDBMS code page. For example, in a Western European code page RDBMS, you can only install Western European language packs such as English, French, Spanish, or German. In a Japanese code page RDBMS, you can only install Japanese or English language packs.

# Upgrade Planning for Siebel Unicode Support

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

For Western European languages and Japanese, Oracle's Siebel applications support both non-Unicode and Unicode code pages. For all other supported languages, Siebel applications support only Unicode code pages.

Unicode is recommended if your installation uses Siebel Email Response, correspondence, or similar functionality, particularly if content is generated on a separate system.

This is because Siebel applications use Unicode internally. If the RDBMS is not using Unicode, Siebel applications convert content from Unicode to the database code page. Using a Unicode code page for the database prevents string conversion problems on text content.

Before converting to Unicode, your encryption method must be RC2 or AES.

**CAUTION:** Migrating to Unicode is more complex than simply importing your existing data into a Unicode database. Failure to execute the migration correctly can result in serious data corruption or unrecoverable data loss. For this reason, Oracle's Siebel Expert Services participation is mandatory. Contact Expert Services to perform a Unicode migration.

For a list of supported languages and code pages, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

For information about Unicode and global deployment for Siebel Business Applications, see *Global Deployment Guide*.

For information on upgrading to RC2 encryption, see *Security Guide for Siebel Business Applications*.

## Planning Considerations for the Unicode Migration

Migrations to Unicode require the assistance of Oracle's Siebel Expert Services.

Contact Expert Services to migrate your upgraded database from a non-Unicode code page to Unicode. To perform the migration, you can use either database vendor native utilities or Siebel utilities.

If you are planning to migrate your upgraded application to Unicode, consider the following points:

- **Database size increase. Migration to Unicode increases 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 Technical Support or Siebel Expert Services.
- **IBM DB2 data truncation.** Migration to Unicode may cause truncation of certain data in DB2 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 are 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 Technical Support or Siebel Expert Services. See also ["Identifying IBM DB2 Long Columns for Truncation in a Siebel Upgrade"](#) on page 165.
- **IBM DB2 custom tablespace information.** The upgrade does not preserve custom tablespace information for DB2 databases. This presents a problem during your migration to a Unicode code page, because you need to know which tables need to be re-created. You must modify the upgrade scripts to handle custom tablespaces.  
  
Contact Siebel Technical Support or Expert Services for instructions about how to modify upgrade scripts to handle custom tablespaces.
- **Third-party product integration.** Migration to Unicode may affect integration with third-party systems. For more information, contact Siebel Technical Support or Siebel Expert Services.
- **Handheld and wireless support.** Certain handheld and wireless application platforms do not support Unicode. For more information, contact Siebel Technical Support or Siebel Expert Services.
- **Environment code pages.** You cannot use a non-Unicode code page for your development environment, and then later migrate to Unicode for your production environment.

## Supported Types of Unicode

The current release supports two types of Unicode:

- **UTF-8.** UTF-8 uses the same encoding for Western European languages. It occupies one byte for Western European languages and up to three bytes for some Asian languages, such as Japanese.
- **UCS-2.** UCS-2 is supported for IBM DB2 and Microsoft SQL Server databases. UCS-2 does not map one-to-one with Western European languages. It occupies two bytes for all languages.

## Upgrade Planning for Siebel RC2 or AES Encryption

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Siebel 7.7 introduced support for the AES encryption method—the government standard for secure applications. Siebel Business Applications continue to support RC2 data encryption.

The default encryption method (called the standard encryptor) is no longer supported. Data that used the standard encryptor cannot be read by applications in the current release. You must upgrade your encryption method to RC2 or AES.

Use the Encryption Upgrade Utility to convert unencrypted data and data that was encrypted using the standard encryptor to the RC2 or AES encryption method. You must run the Encryption Upgrade Utility even if you are upgrading to stronger encryption, for example, from RC2 56-bit to RC2 128-bit encryption. Upgrading to RC2 128-bit or AES encryption requires the Siebel Strong Encryption Pack.

To upgrade your encryption method, see *Security Guide for Siebel Business Applications*.

## Upgrade Planning for Siebel Web Template Files and Style Sheet

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

Web template files help define the layout and formatting of the user interface, such as views, applets, and controls. The Siebel Web Engine in the Siebel Server uses the Web template files to build Web pages, which it then forwards to the Web Server.

A Web template file contains regular HTML, WML, or XML tags interspersed with Siebel tags. Siebel tags are prefixed by “swe” and contain placeholders for UI objects such as controls and data. HTML formatting tags are defined in a cascading style sheet (main.css).

When you install a new release of the Siebel Server or Siebel Tools, you receive a new set of Web template files and a new style sheet file. The upgrade process does not use your existing files. If you have manually customized your Web template files or style sheet file, you must evaluate whether to reimplement these customizations in the new Web template files and style sheet file.

Observe the following planning guidelines for reimplementing customizations:

- Resolve any UI problems related to object definitions in the Siebel Tools repository first. While doing so, review the areas of the new UI where you plan to implement Web template file customizations.
- Evaluate existing customizations to Web template files, and decide which ones to reimplement. Changes to the UI in the new release may make some customizations obsolete.
- Document why each customization was reimplemented. This reference will help you evaluate customization issues later.
- Use formal change control for managing versions of the Web template files and style sheet file. This allows you to maintain orderly distribution of the files to developers.
- Applets typically have a separate Web template file for each applet mode. Customize all the mode Web template files for an applet at the same time. This allows you to verify applet functionality in a single test pass in the UI.
- Individual Web template files are typically used by multiple screens, views, or applets. Set up formal test plans that verify customizations are correct across all UI objects that use each Web template file. This reduces the amount of time required to verify customizations and prevents unintended changes to the UI.
- Reimplement style sheet customizations in two passes. On the first pass implement only those changes required to UI usability issues. On the second pass, implement the remaining style sheet customizations after Web template customizations are complete. This shortens the time required to resolve functional problems in the UI.

After customizations have been completed, the Web template files and style sheet file must be copied to the Siebel Servers in the environment. The files must also be included in upgrade kits sent to remote sites.

Web template files have an `.swt file` extension and are located in the `webtempl` directory in both the Siebel Tools and Siebel Server installation.

The style sheet file is located as follows in the installation directory of Siebel Tools and the Siebel Server (Windows path syntax):

Siebel Tools: `\publ i c\lang\files\main.css`

Siebel Server: `SIEBEL_ROOT\webmaster\files\lang\main.css`

where *lang* is the installation language, for example, `enu`.

The steps for reimplementing customizations to these files and copying them to new environments are included in the upgrade process topics in [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

For information on how Web template files and style sheet file work, see *Configuring Siebel Business Applications*.

# Upgrade Planning for Siebel Access Control

**Upgrades from:** Siebel 6.x & 7.0.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Access control was significantly revised in Siebel 7.5. Access control refers to all mechanisms that control visibility of screens, views, and data within Siebel Business Applications. Access control includes, but is not limited to positions, responsibilities, organizations, and access groups.

To implement access control within your Siebel Business Applications, your Siebel administrator creates relationships between people and resources (a more general term for data that includes views and functionality). These relationships or policies are authorizations. Both people and resources can be grouped and placed in hierarchies to simplify administration.

External users, such as customers and channel partners, can be assigned varying access levels that control visibility of data and application functionality. When planning access policies, consider the following:

- 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).
- The amount of content that is distributed by the Siebel Business 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).
- The number of users and entities that access the data. Also consider the complexity of relationships between users (partners, competitors, browsers, customers).

For more information on access control, see *Security Guide for Siebel Business Applications*.

## Person, Household and Service Request Visibility

Beginning with Siebel 7.5, Person, Household, and Service Request can be made visible to multiple organizations, also called Business Units. Siebel 7.5 introduced several new tables to support this:

- S\_CONTACT\_BU
- S\_ORG\_GROUP\_BU
- S\_SRV\_REQ\_BU

The upgrade to Siebel 7.7 populates the S\_CONTACT\_BU, S\_ORG\_GROUP\_BU, and S\_SRV\_REQ\_BU tables with one record for each record in the S\_CONTACT, S\_ORG\_GROUP, and S\_SRV\_REQ tables. After the upgrade, Contacts, Households, and Service Requests continue to be visible from the Business Unit they belonged to before the upgrade.

## Access Group and Userlist Attributes

In Siebel 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 to Siebel 7.7 adds records to the S\_PARTY\_GROUP and S\_USERLIST tables for existing S\_PARTY Access Group and User List records.

To support Multi-Org visibility, the upgrade also adds corresponding intersection table records to the S\_PARTY\_GRP\_BU and S\_USERLIST\_BU tables.

Technical Note 312 provides guidance and best practices for implementing access control. This Technical Note includes background information about the Access Group access control mechanism implemented in Siebel 7, discusses migration considerations, and outlines steps for deploying Access Group access for Siebel Business Applications.

For detailed information about access control, see *Security Guide for Siebel Business Applications*.

## Content Categorization

Product categorization was available in the Siebel 6.x data model. Siebel 7.x expands this to support categorization of content such as auction items and literature items.

Categorizing content simplifies access control policy design and management. System administrators can specify access to a set of content items. This makes the content more searchable and accessible to users through navigation.

In Siebel 6.x, categories could be shared across multiple catalogs and could have multiple parents. In Siebel 7.x, a category can belong only to one catalog and have at most one parent catalog.

To accommodate this change, the database upgrade generates copies of categories and category hierarchies that were previously shared across multiple catalogs.

For example, if Catalog-A and Catalog-B both share Category-1, then after the upgrade, Category-1 will not be shared. Instead, there will be two copies of Category-1. One copy will belong to Category-A. The other copy will belong to Category-B.

For more information about categorization, see *Security Guide for Siebel Business Applications*.

# Upgrade Planning for Siebel Database Clustered Indexes

**Upgrades from:** Siebel 6.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

If you created clustered indexes on base tables, and the upgrade introduces a different clustered index on the same table, the upgrade process re-creates custom indexes as nonclustered and creates the Siebel index as clustered.

For IBM DB2, indexes that reside on tables with append mode enabled are re-created as nonclustered indexes during the upgrade. (Tables created with append mode enabled do not support clustered indexes.)

# 6

## Application Planning for a Siebel Upgrade

This chapter contains the following topics:

- [“Determining Your Siebel 7.5.x Release Level” on page 137](#)
- [“Upgrade Planning for Migrating Siebel Address Data” on page 139](#)
- [“Upgrade Planning for Siebel Employee Relationship Management \(ERM\)” on page 140](#)
- [“Upgrade Planning for Siebel Marketing” on page 141](#)
- [“Upgrade Planning for Siebel Workflow Designer” on page 147](#)
- [“Upgrade Planning for Handheld Devices in the Siebel Environment” on page 148](#)
- [“Upgrade Planning for Resonate Central Dispatch in the Siebel Environment” on page 149](#)
- [“Upgrade Planning for Siebel String Translation” on page 149](#)
- [“Upgrade Planning for Siebel Configurator” on page 149](#)
- [“Upgrade Planning for Siebel Personalization” on page 150](#)
- [“Upgrade Planning: Additional Siebel Application Changes” on page 150](#)

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Determining Your Siebel 7.5.x Release Level

**Upgrades from:** Siebel 7.5.x.

**Environments:** Development environment only.

Before upgrading your enterprise to the new release, determine your 7.5.x release level.

When you run the Database Server Configuration Utilities to perform the upgrade, they prompt you to choose which release you are upgrading from. The choices include: 7.5.2, 7.5.2.200+, and 7.5.3.

If you choose the wrong 7.5.x release level, the repository merge will complete successfully, but you may have UI configuration problems.

The release level of an installed release is contained in the base.txt file located in directories throughout the enterprise installation.

**CAUTION:** Perform the following tasks before upgrading enterprise components to the new release. Installing the new release replaces the base.txt files in the old release.

## Determining Your Release Level

Perform this task to identify your 7.5.x release level.

### *To determine your release level*

- 1 On the machine where the Siebel Server you will use to perform the upgrade is installed, locate the Siebel Server installation directory, siebsrvr.
- 2 In the siebsrvr directory, open base.txt and write down the release level.  
For example, in the following base.txt string, the release level is 7.5.3:  

```
7.5.3 SIA [XXXXX] LANG_INDEPENDENT full release
```
- 3 Store the 7.5.x release level where you can refer to it when you run the Database Server Configuration Utilities to upgrade the Siebel database.

## Verifying the Repository Update Is Installed

Perform this task if your release level is 7.5.2.200+ or 7.5.3.

The 7.5.2.200+ and 7.5.3 maintenance releases included a repository update that must be installed manually. Verify that this update is installed. If you have not installed the update, refer to the *Maintenance Release Guide for Siebel 7.5.3* on Siebel SupportWeb and install the update.

### *Verifying the repository update is installed*

- 1 Open Siebel Tools.
- 2 In the Object Explorer, select Applet.
- 3 Query for Account Team Mvg Applet, and select this applet.
- 4 In the Object Explorer, select Control, a child object of Applet.
- 5 With the Controls list active, query for \*shuttle.
- 6 Verify the following controls are included in the query returns:
  - Add - Shuttle
  - AddAll - Shuttle
  - Remove - Shuttle
  - RemoveAll - Shuttle
  - LabelSelected - Shuttle
- 7 Choose one of the following actions:
  - If all the controls are present, the repository update is installed. No further action is required.
  - If none of the controls are present, the repository update is not installed. Refer to the *Maintenance Release Guide for Siebel 7.5.3* on Siebel SupportWeb and install the update.
  - If some of the controls are present, but not all, contact Siebel Technical Support.

# Upgrade Planning for Migrating Siebel Address Data

## Upgrades:

- From Siebel Industry Applications (SIA) 6.x, (except Communications, Media, and Energy) to Siebel SIA 7.7.x
- From Siebel Financial Services 6.x and 7.0.x to Siebel SIA 7.7.x
- From Siebel Business Applications 6.x and 7.x to Siebel Business Applications 7.7.x as part of upgrading to Siebel SIA 7.7.x
- From Siebel Financial Services 6.2.1 on IBM z/OS platforms to Siebel SIA 7.7.x on IBM z/OS platforms

**NOTE:** This topic does not apply to Siebel Business Applications (HOR) that you are upgrading to a later release of Siebel Business Applications (HOR).

Siebel 7.7 (and Siebel 7.5 SIA) changed the way address information is stored. You must perform several tasks to ensure address migration is handled correctly.

## Previous Releases

In previous releases, address data was stored as follows:

- The relationship between person and address was 1:M and was stored in the table S\_ADDR\_PER.
- The relationship between account and address was 1:M and was stored in S\_ADDR\_ORG.
- Both tables included a column ADDR\_NAME, which is a computed value based on other attributes in the address table.
- The user key for S\_ADDR\_PER included PER\_ID and ADDR\_NAME.

## Siebel 7.7

In Siebel 7.7, the relationship between person and address and between account and address is M:M. Address information is stored in S\_ADDR\_PER. The relationship between addresses and contacts and between addresses and organization is stored in S\_CON\_ADDR.

The upgrade process revises storage of address data as follows:

- Inserts data into S\_CON\_ADDR from S\_ADDR\_PER and S\_ADDR\_ORG.
- Migrates data from S\_ADDR\_ORG to S\_ADDR\_PER. The table S\_ADDR\_ORG is obsolete.
- Sets S\_ADDR\_PER.PER\_ID to NULL. It is no longer used as part of the user key but remains part of the physical key.
- Uniqueness of addresses in S\_ADDR\_PER is enforced only on ADDR\_NAME.
- The table S\_CON\_ADDR becomes the intersection table for data stored in S\_ADDR\_PER and account or contact data stored in S\_ORG\_EXT or S\_CONTACT respectively.

## How Address Data Is Preserved

Because PER\_ID is no longer part of the user key for S\_ADDR\_PER, the ADDR\_NAME must be unique for all records.

It is possible that records within or across S\_ADDR\_ORG and S\_ADDR\_PER could have the same ADDR\_NAME. If this occurs, the ADDR\_NAME for one of the records is preserved, and the upgrade process appends the ROW\_ID to ADDR\_NAME for the others. This prevents records from being deleted and preserves all records from both tables.

## How to Manage Address Migration

You must perform the following tasks to migrate address data:

- Before upgrading the database, you must run a script to identify records that have the same ROW\_ID between S\_ADDR\_PER and S\_ADDR\_ORG. You must eliminate duplicate row IDs.
- You must evaluate whether to modify upgrade scripts to migrate address data in custom extension columns in S\_ADDR\_PER and S\_ADDR\_ORG. Review the new schema and determine if your custom columns can be mapped to standard columns. If no matching standard columns exist, create new columns on the target tables. During the database upprep, you do this after running the Database Server Configuration Utilities but before running the Upgrade Wizard.
- After the upgrade is complete, review the records in S\_ADDR\_PER and eliminate duplicate and obsolete records.

To manage address migration, follow the steps in [“Process of Upgrading a Siebel Production Test Environment” on page 47](#). Each of the address migration tasks is included as a step in this process. Each step refers you to a procedure for performing the task.

# Upgrade Planning for Siebel Employee Relationship Management (ERM)

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Siebel 7.7 ERM introduced important changes to workflows, Siebel Training, Microsite management, and Group News.

## Workflows

Siebel 7.5.3 included approval business process workflows in the Sample database. If you activated these workflows, or created workflows containing approval steps, you must manually upgrade these workflows.

## Siebel Training

In previous releases, the product catalogs containing Siebel Training courses were of type Buying. Siebel 7.7 provides a new catalog type called Training. If you have catalogs that contain both Siebel Training products and other kinds of products, revise these catalogs so that they contain only Siebel Training products. This prevents nontraining products from being moved to the training catalog during upgrade.

## Microsite and Group News

If you have created customized microsite or Group News page sections, you must revise the associated business component and applet definition.

# Upgrade Planning for Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Siebel 7.7 introduced three architectural changes to Siebel Marketing:

- **Simplified hierarchy.** Program occurrences and campaign occurrences have been removed. This simplifies the Siebel Marketing hierarchy.
- **Merged Business Objects.** In previous releases, there were two business objects that supported similar campaign functions, DBM Campaign and Campaign. Siebel 7.7 merged all campaign administration functions into the Campaign business object.
- **Marketing Server Migrated to Siebel Analytics.** The Marketing Server is obsolete. Its functions have been migrated to Siebel Analytics and are implemented on the Siebel Analytics Server.

## Simplified Hierarchy

The Program Plan > Campaign Plan > Campaign Occurrences hierarchy has been simplified. The new hierarchy is Programs > Campaigns > Waves.

During the upgrade, objects that were related to the campaign occurrence are re-parented to the surviving campaign. Child objects are re-parented from a campaign occurrence to a campaign, as shown in Figure 4.

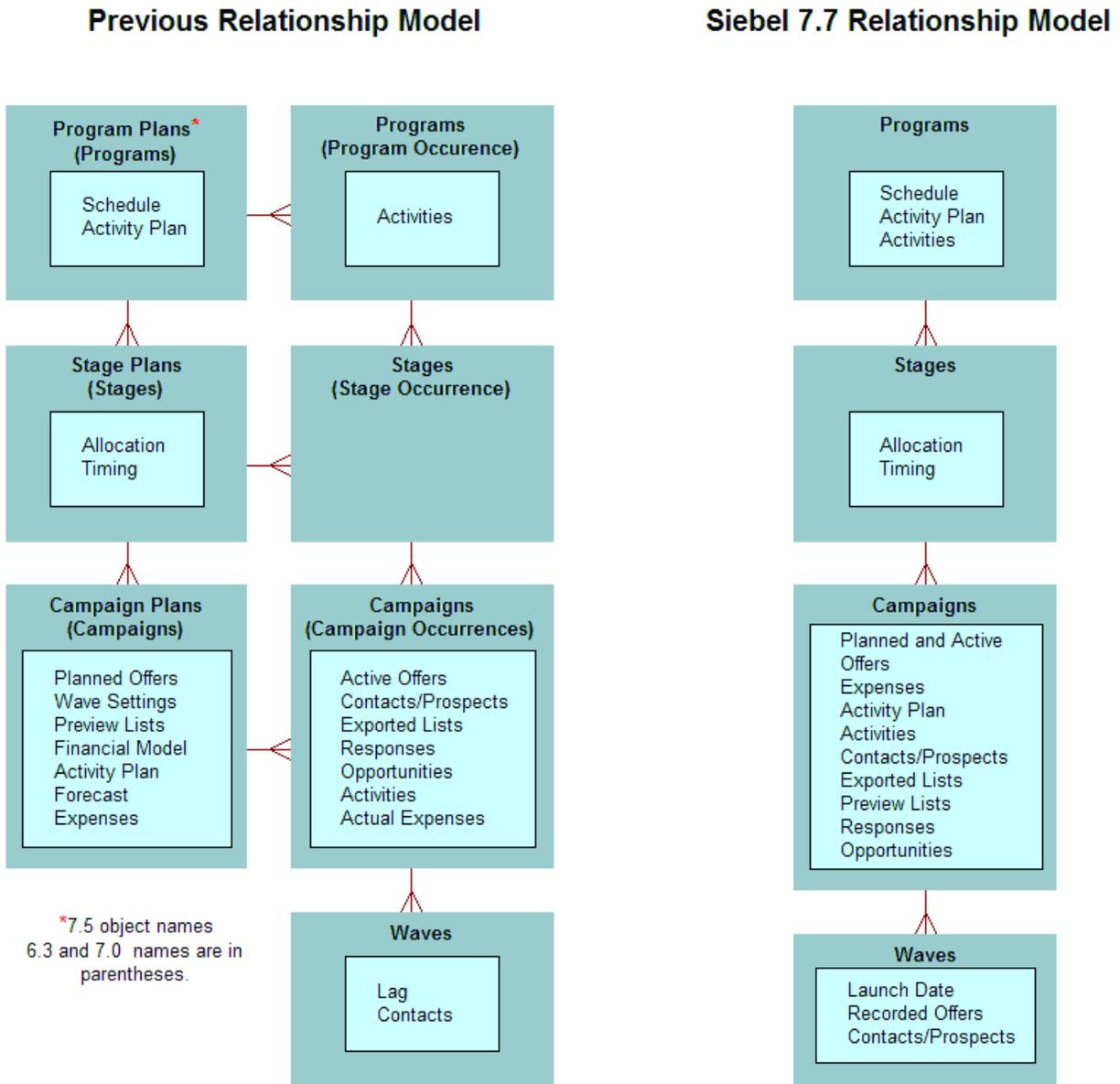


Figure 4. Relationship Model for Siebel Marketing

## Merged Campaign Business Objects

In previous releases, campaign functions were handled by two business objects—DBM Campaign and Campaign. These objects shared similar functions, but many related elements such as views and child objects were not shared. The upgrade merges all campaign functions and views into a single business object—the Campaign business object.

During the upgrade, the following changes are performed to merge campaign plans and campaigns:

- Campaign Plans from the previous release are converted into Campaigns.
- The campaign occurrences that were children of the Campaign Plan are made obsolete.
- A new Load record is created for each previous occurrence.

Any child objects that were related to the obsolete campaign occurrence are re-parented to the surviving campaign.

## Marketing Server Migrated to Siebel Analytics

The upgrade obsoletes the Data Dictionary Manager and Marketing Server components in the Siebel Server. In previous releases, the Marketing Server component group performed segmentation and list export. Segmentation and list export are handled by subsystems in the Siebel Analytics Web server.

Due to the obsolete server components, a set of business components and other objects are also made obsolete. These objects remain in the repository but are no longer used by the application.

The upgrade does not obsolete the List Import Service Manager, Marketing AOM (SMObjMgr), eMarketing AOM (eMarketObjMgr), or the eEvents AOM (eEventsObjMgr). They are actively used in the application.

[Appendix A, “Siebel Marketing Upgrade Reference”](#) lists the following types of obsolete items:

- Server components
- Business objects and business components
- Screens and views
- Data that is obsolete

## About Siebel Marketing Customizations

Siebel 7.7 introduced a feature, Incorporate Custom Layouts (ICL). When you select this feature in the Upgrade Wizard, customized view and applet layouts are preserved, consistent with new features and schema changes.

Because of the large number of user interface changes in Siebel Marketing, the views and applets for Siebel Marketing have been excluded from ICL support. (The Upgrade Behavior property for Siebel Marketing objects in the Siebel Tools Repository is set to ADMIN.)

This means that the Repository upgrade will not preserve customizations to Siebel Marketing views and applets. If you select ICL mode for your upgrade, then other views and applets (except those for Siebel Marketing) will be preserved under ICL.

## Summary of Database Changes for Siebel Marketing

Table 34 on page 144 summarizes Siebel database changes that apply to Siebel Marketing.

Table 34. Summary of Database Changes for Siebel Marketing for Siebel 7.7

Upgrade Step	Description
Updates S_SRC and S_SRC_REL	Updates marketing plan tactics to use the S_SRC.MKTG_PLAN_ID to indicate the parent marketing plan, rather than S_SRC_REL.
Updates S_SRC	<p>Updates Status values (STATUS_CD) for campaigns and programs.</p> <p>Updates campaign plans and stand-alone campaigns to use the same CAMP_TYPE_CD.</p> <p>Sets CAMP_TYPE_CD to Campaign when SUB_TYPE is Marketing_Campaign and other conditions specified in upgrade scripts are met.</p> <p>Updates the values for S_SRC.RESPNSE_TYPE.</p>
Inserts into S_DD_CAMP_WAVE and S_CAMP_LD_WAVE	<p>Creates a Load record for each campaign occurrence. Load numbers are assigned based on date of execution.</p> <p>Creates one load wave for each stand-alone campaign.</p> <p>Sets the status of the load wave to the corresponding execution status from the original campaign occurrence.</p>
Inserts into S_CAMP_WAVE_DCP	Records the offer history for load wave records (pre-upgrade occurrences).
Index change and update to S_CAMP_CON	<p>Modifies the user key to use Campaign ID + Contact ID + Load Number + Token Number.</p> <p>When campaign occurrences are converted to Load records, S_CAMP_CON is updated to the appropriate Load number for the campaign contact.</p>
Updates S_SRC_COST	Updates TYPE_CD to the new values for Fixed and Per Unit expenses.
Updates S_SRC_GOAL	Updates goal type codes (GOAL_TYPE_CD).
Updates S_EVT_ACT	<p>Re-parents activities from campaign occurrences to campaigns.</p> <p>Re-parents activities from program occurrences to programs.</p>

Table 34. Summary of Database Changes for Siebel Marketing for Siebel 7.7

Upgrade Step	Description
Updates S_SRC for Stages	Updates CAMP_TYPE_CD = "STAGE" for stage records.
Updates S_OPTY_SRC	Re-parents opportunities from campaign occurrences to campaigns.
Updates S_SRC_DCP	Re-parents offers from campaign occurrences to campaigns.
Updates S_ORDER	Re-parents orders from campaign occurrences to campaigns.
Updates S_CS_PATH_SRC	Re-parents SmartScripts from campaign occurrences to campaigns.
Updates S_SRC_POSTN	Re-parents team members from campaign occurrences to campaigns.
Updates S_COMMUNICATION	Re-parents responses from campaign occurrences to campaigns.
Updates S_CAMP_SKILL	Re-parents Assignment Skills from campaign occurrences to campaigns.
Updates S_QTA_PLAN	Re-parents Quota Plans from campaign occurrences to campaigns.
Updates S_PROD_INT_SRC and S_PROD_LN_SRC	Re-parents related products and product lines from campaign occurrences to campaigns.
Updates S_CALL_LST	Re-parents internal lists from campaign occurrences to campaigns.
Inserts into S_SRC_GOAL (for Campaign Plan)	Updates actual and forecast values for Goals based on Financial Modeler columns in S_SRC (Revenue, #Leads, Response Rate, Conversion Rate, Avg. Revenue per Sale, Avg. Contribution Margin, ROI Amount).

### Summary of How Marketing Data Will Be Migrated

Table 35 on page 146 describes how Siebel Marketing data will be migrated.

Data that is obsolete as of Siebel 7.7 is listed in [Appendix A, "Siebel Marketing Upgrade Reference."](#)

Table 35. Siebel Marketing Data Migration for Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Marketing Plans	Preserved	N/A	N/A	
Program Plans	Preserved	N/A	N/A	
Stage Plans	Preserved	N/A	N/A	
Campaign Plans	Preserved	N/A	N/A	Campaign Plans are converted to Campaigns.
Campaign History	Preserved	N/A	N/A	Index on S_CAMP_CON is modified to preserve history for campaign and campaign load.
Stand-alone Campaigns (campaigns not related to a program)	Preserved	N/A	N/A	
Response History	Preserved	N/A	N/A	
Lists	Preserved	N/A	N/A	
Offers (except eNewsletter offers)	All offer types preserved, except for eNewsletters	See Obsolete Objects table	See Obsolete Objects table	
Segments	Usage history in programs is preserved  Segment criteria are obsolete	S_CALL_LST_CRIT  S_CALL_LST_DTL  S_CALL_LST_QRY	Segment Detail-DD  Segment Expression  Campaign Segment Allocation  Segment Campaign Allocation	Segments should be reconstructed in the Siebel 7.7 Segment Designer.

# Upgrade Planning for Siebel Workflow Designer

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Important changes were made for Siebel 7.7 to how workflows are upgraded. Use this topic to determine how these changes affect your upgrade.

## Changes for Siebel 7.7

In Siebel 7.7, the Workflow Process Designer was moved to Siebel Tools. Workflow components and definitions are defined as Siebel Tools objects and are stored in the repository.

The upgrade script migrates workflows from the main Siebel database to the repository. Before upgrading your development or test environments, make sure that they include all the workflows from your production environment.

The upgrade copies or moves all workflow definitions to the repository as follows:

- **Seed workflows.** The old seed workflows are overwritten by new seed workflows. Their status is Inactive.
- **Customer workflows, status Inactive.** These are converted to workflow definitions in the repository. They will not have a status.
- **Customer workflows, status In-Progress.** These are converted to workflow definitions in the repository. Their status remains In-Progress.
- **Customer workflows, status Active.** These are converted to workflow definitions in the repository. Their status is changed to Completed. These workflow definitions are not copied to the main Siebel database. This means that after the upgrade, no workflows are deployed. You must manually deploy seed workflows and customer workflows after the upgrade.

The upgrade automatically migrates customized business service scripts that are called by workflow operations.

You must deploy and activate repository workflows in order to use them. Workflow policy object and policy program data is upgraded normally. No data is changed or lost. Database triggers are not upgraded. After the upgrade, you must regenerate database triggers.

For information on how to deploy workflows, see *Siebel Business Process Designer Administration Guide*.

## Schema Changes

The main Siebel database tables that contain workflow definitions have changed. The new tables contain the workflow definitions for deployed workflows. The definitions of workflows that are Inactive or In-Progress are located in the repository. The tables are named as follows:

- Siebel database tables containing workflow information begin with S\_WFA.
- Repository tables containing workflow information begin with S\_WFR.

- Siebel database tables that contain workflow information for releases prior to Siebel 7.7 begin S\_WF\_ (note the underscore after WF). After the upgrade to Siebel 7.7, these tables are obsolete and are not referenced by applications.

## Upgrade Planning for Handheld Devices in the Siebel Environment

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Device operating system support, third-party product support, and application configuration management for handheld devices were revised in Siebel 7.7. Before the upgrade, verify that handheld devices are running an OS supported by Siebel 7.7. Also, verify that third-party software is the correct version.

After the upgrade, you must enter any handheld device-related application configuration changes into the handheld device administration screen.

### Supported Device OS

The 2002 version of the Pocket PC operating system is no longer supported as of Siebel 7.7. Devices running Pocket PC 2002 must be upgraded to Pocket PC 2003. For a full description of supported handheld devices and operating systems, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

### Active Sync Support

The third-party product Active Sync must be at version 3.7.1 or higher as of Siebel 7.7. For a complete description of supported third-party products, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

### Handheld Device Administration

Siebel 7.7 introduced a handheld device administration screen. If you have modified the PDA portion of the Siebel Sales, Siebel Service, Siebel ePharma, or Siebel eConsumer application .cfg file, do the following before upgrading:

- Save the modified .cfg file under a new name so that it is not overwritten during the upgrade.
- After the upgrade, go to the Administration—Mobile screen and enter the changes from the saved .cfg file. For information on using this screen, refer to the Handheld guide for your application.

### Handheld Application Upgrade

Handheld applications do not upgrade automatically. Users must uninstall the application and install the current release.

## Upgrade Planning for Resonate Central Dispatch in the Siebel Environment

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Support for Resonate Central Dispatch is discontinued as of Siebel 7.7. It has been replaced by a load balancing module that is included in the Siebel Web Server Extension. In addition, Oracle has certified several third-party HTTP load balancers for use with the Siebel Web Server Extension.

For a description of the load balancing module, see *Deployment Planning Guide*. For a list of supported HTTP load balancers, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

For information on configuring load balancing, see *Siebel Installation Guide* for the operating system you are using.

## Upgrade Planning for Siebel String Translation

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

Siebel 7.7 introduced a new method of generating data-migration SQL. When the SQL commands for the upgrade are created, they contain generic values or placeholders. A set of files maps these values to language-specific strings. The upgrade process substitutes these strings in the SQL commands. The translated strings typically are those associated with seed data such as menu items in LOVs.

The strings assigned to values are based on those used in previous Siebel releases. If you have made changes to string mappings in a previous release, contact Siebel Technical Support or Siebel Expert Services for guidance on managing string mapping during the upgrade.

**CAUTION:** If you have modified string mappings in a previous release, you must obtain Oracle assistance before upgrading. If you do not do so, data may be lost or corrupted during upgrade.

## Upgrade Planning for Siebel Configurator

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

At Siebel 7.7, you can set a user property in a Siebel Configurator model that causes configuration sessions to run in high interactivity mode. This feature improves performance and is intended for use with Oracle's Siebel applications that are typically deployed in high interactivity mode, such as employee applications.

For information on how to deploy Siebel Configurator models in high interactivity mode, see Technical Note 500 on Siebel SupportWeb.

Browsers that will host high interactivity applications must meet certain requirements. These are described in the *System Requirements and Supported Platforms* on Siebel SupportWeb. For more information on high interactivity mode, see the *Deployment Planning Guide*. If you intend to deploy Siebel Configurator models in high interactivity mode, include verifying that users have appropriate browsers in your upgrade planning.

## Upgrade Planning for Siebel Personalization

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

At Siebel 7.7 access control based on primary user role has been replaced by access based on primary responsibility name. If you have defined conditional expressions for applets in the Administration-Personalization screen, plan to review these after the upgrade and verify that they use Primary Responsibility name. For more information on access control, see *Security Guide for Siebel Business Applications*. For more information on personalization management, see *Siebel Personalization Administration Guide*.

## Upgrade Planning: Additional Siebel Application Changes

**Upgrades from:** All Siebel releases.

[Table 37](#) displays additional application changes. Review these changes and make needed revisions before beginning the upgrade.

Table 37. Additional Application Changes

Applications or Views Affected	Applies to Upgrades from	Description
eCalendar Detail View	Siebel 7.5 and earlier	In Siebel 7.7, the HI Gantt Chart Applet is no longer used. It is replaced by the Calendar Gantt Chart AX Applet.

Technical Note 511 on Siebel SupportWeb lists application objects in the Siebel Repository that are obsolete. Recommended actions are listed for each object, if applicable. Before upgrading, review this list and determine if any of your customizations are using objects that obsolete as of the new Release.

# 7

## Basic Database Preparations for a Siebel Upgrade

This chapter contains the following topics:

- [“Verifying Siebel Database Connectivity” on page 151](#)
- [“Preparing Siebel Tables and Views for Upgrade” on page 151](#)
- [“Preparing Siebel Custom Indexes for Upgrade” on page 152](#)
- [“Exporting Siebel Interface Table Data” on page 153](#)
- [“Archiving Unneeded Siebel Repositories” on page 153](#)
- [“Preserving Siebel Dock Objects and Visibility Rules” on page 154](#)
- [“Securing AIX Memory Allocation Segment Space for the Siebel Database” on page 154](#)

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Verifying Siebel Database Connectivity

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

From the production test environment, you must be able to make ODBC connections to both the Siebel database in the development environment and the Siebel database in the production environment. Verify that you can define these ODBC connections in the production test environment.

If you cannot connect to these databases from the production test environment, contact Siebel Technical Support.

In the production environment, you do not have to define an ODBC connection to the development environment Siebel database or the production test environment Siebel database.

## Preparing Siebel Tables and Views for Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

### *To prepare tables and views for upgrade*

- 1 Drop temporary tables and non-Siebel tables.

If the upgrade process detects a column with a datatype not acceptable to Siebel tables, the upgrade will fail.

- 2 Disable customized triggers.

You must re-create them after the upgrade.

- 3 Drop defined database views on Siebel tables.

You must re-create them after the upgrade.

- 4 Export interface table data that you want to preserve.

Interface tables are dropped and then re-created during upgrade. You can import the data after the upgrade.

## Preparing Siebel Custom Indexes for Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Consider the following guidelines when preparing custom indexes for upgrade:

- **Custom indexes against extension columns on obsolete tables.** If you have created custom indexes that use extension columns on obsolete tables, you must migrate the data to new extension columns before upgrading the Siebel database. For assistance, contact Siebel Technical Support.
- **Custom indexes that were not defined through Siebel Tools.** Custom indexes created without using Siebel Tools are not included in the schema definition in the Siebel Repository. These indexes are dropped during the database upgrade. To preserve these indexes, add them to the Siebel Repository using Siebel Tools.
- **Custom indexes on interface tables.** Custom indexes on interface tables are not re-created during the upgrade. You must re-create them after the upgrade is complete.

- **Custom indexes on base tables.** The Siebel 7.x upgrade automatically drops and re-creates custom indexes on base tables.
- **Custom indexes may need to be changed to reflect schema changes.** Reevaluate custom indexes for applicability in the new release. They may no longer be needed due to schema changes in the new release.

For more information about custom indexes, see *Configuring Siebel Business Applications*. For information on schema changes in a release, see *Siebel Data Model Reference*.

## Exporting Siebel Interface Table Data

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

During the upgrade process, your interface tables are dropped and then re-created. To retain data in your interface tables, use the appropriate tools for your RDBMS to export data before the upgrade and then import the data after you have completed the upgrade.

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

## Archiving Unneeded Siebel Repositories

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Perform this task before running the Database Server Configuration Utilities in upgrep mode for the first time in an environment.

The upgrep mode scripts expect only the Siebel Repository to be present. If you have multiple repositories, you must export them to archive files and then delete them from the database.

Because the upgrade changes the schema of the database, in most cases you cannot import these archived repositories into the upgraded database. If you want to access the archived repositories, you must import them into a database that has the same schema as the one from which they were exported.

### *To archive unneeded repositories*

- 1 Export all repositories.

- 2 Place exported repository files in a safe location.
- 3 In the Siebel database, delete all repositories except the Siebel Repository.

For information on exporting and deleting repositories, see the Managing Repositories chapter in *Using Siebel Tools*.

## Preserving Siebel Dock Objects and Visibility Rules

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Modified visibility rules are dropped during a development environment upgrade. Manually record any changes to dock object visibility rules, so you can evaluate whether you must reapply the changes after the upgrade is complete.

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

Changing the definition of dock objects requires the assistance of Siebel Technical Support or Siebel Expert Services.

## Securing AIX Memory Allocation Segment Space for the Siebel Database

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases. (Exception: this topic does not apply to Microsoft SQL Server.)

**Platforms:** UNIX only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Before you run an upgrade on AIX, set the following environment variable on the AIX machine that you are using for the upgrade:

```
setenv LDR_CNTRL LOADPUBLI C@MAXDATA=0x60000000
```

This will prevent a shortage of memory allocation segment space that might occur on the machine where both the Siebel Database Server and Siebel Server are installed. After a successful upgrade, reset this parameter to the original value.





# 8

## Preparing an IBM DB2 Database for a Siebel Upgrade

This chapter contains the following topics:

- [“Verifying the IBM DB2 Client for a Siebel Upgrade” on page 157](#)
- [“Verifying IBM DB2 Sort Order for a Siebel Upgrade” on page 158](#)
- [“Setting IBM DB2 Parameters for a Siebel Upgrade” on page 158](#)
- [“Verifying IBM DB2 Permissions for a Siebel Upgrade” on page 160](#)
- [“Verifying IBM DB2 Instance Owner Permissions for a Siebel Upgrade” on page 161](#)
- [“Creating IBM DB2 Temporary Tablespaces and Bufferpools for a Siebel Upgrade” on page 161](#)
- [“Analyzing IBM DB2 Custom Tablespace Requirements for a Siebel Upgrade” on page 162](#)
- [“Verifying the DB2 Application Development Client for a Siebel Upgrade” on page 164](#)
- [“Identifying IBM DB2 Long Columns for Truncation in a Siebel Upgrade” on page 165](#)
- [“Installing IBM DB2 Stored Procedures and Functions for a Siebel Upgrade” on page 166](#)

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Verifying the IBM DB2 Client for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only. This topic applies only to the 64 bit DB2 RDBMS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Siebel Server supports only the 32-bit DB2 client. Verify that you have not installed the 64-bit DB2 client on the Siebel Servers. If you have installed the 64-bit DB2 client, replace it with the 32-bit client.

IBM supports the 32-bit DB2 client working with the 64-bit DB2 8.x RDBMS.

## Verifying IBM DB2 Sort Order for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Binary sort order is required for the development environment upgrade and is strongly recommended for the production environment upgrades.

Sort order is specified during creation of the database. If you find that your IBM DB2 development database was not created using Identity sort order, you must re-create your database using the option `COLLATE USING IDENTITY`.

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

### *To verify that your database was created using Identity sort order*

1 Run the following query on Siebel database:

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

2 Review the result.

■ If sort order is correct, the result is

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

■ If sort order is incorrect, you must re-create the database, using this option:

```
COLLATE USING IDENTITY
```

### Related Topics

[“About Database Sort Order in the Siebel Environment” on page 123](#)

## Setting IBM DB2 Parameters for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Before upgrading an IBM DB2 database, verify that your database server meets or exceeds the following configuration criteria:

- The DMS tablespace has at least 25% of free pages.
- The file system has sufficient space to allow your DMS tablespace to grow.
- Siebel tablespaces for the DB2 platform should be database-managed tablespaces (DMS) rather than system-managed tablespaces (SMS).
- Verify that the tablespaces are not near their capacity. This can be done by connecting to the database and issuing the following command:

```
db2 list tablespaces show detail
```

The tables below provide upgrade-specific settings for the Database Manager and database. Use the following strategy to set parameters:

- Set parameters using the recommendations in *Siebel Installation Guide* for the operating system you are using. Recommendations are located in the chapter on configuring the RDBMS.
- For the upgrade, revise the configuration parameters listed below.
- After the upgrade, reset the configuration parameters to the values listed in *Siebel Installation Guide* for the operating system you are using.

## DB2 Database Manager Settings

The Upgrade Setting column in [Table 38](#) provides guidelines for setting configuration parameters specifically to optimize upgrade performance. Set these parameters for each DB2 instance.

Table 38. DB2 Database Manager Configuration Parameters

Parameter	Explanation	Upgrade Setting
SHEAPTHRES	Sort heap threshold (4 KB)  If you reset SHEAPTHRES or SORTHEAP, rebinding the instance is recommended.	Double the value allocated for SORTHEAP. See <a href="#">Table 39 on page 160</a> .

## DB2 Database Configuration Parameters

The Upgrade Setting column in [Table 39](#) provides guidelines for setting configuration parameters specifically to optimize upgrade performance. Set these parameters for each DB2 instance.

Table 39. DB2 Database Configuration Parameters

Parameter	Explanation	Upgrade Setting
SORTHEAP	Sort list heap (4 KB)	20000–40000 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 20000 setting allows SORTHEAP to increase up to 80 MB.
MAXLOCKS	Percentage of lock lists per application	5
CHNGPGS_THRESH	Changed pages threshold	5
LOGRETAI N	Sequential or circular log files	NO  Setting this parameter to YES means that log files are archived. You must periodically move or archive the log files to prevent the file system containing the log files from filling up.
LOGFILSI Z	Log file size (4 KB)	Development environments: 8000–16000
SOFTMAX	Triggers bufferpool flushing	50

## Verifying IBM DB2 Permissions for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

**Platforms:** UNIX only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you are running IBM DB2 on the AIX or Solaris platform, perform the following steps before executing the Siebel database upgrade.

### *To verify DB2 permissions*

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

```
ls -ld sql lib/function/routine/sql proc
```

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

```
chmod g+w sql lib/function/routine/sql proc
```

## Verifying IBM DB2 Instance Owner Permissions for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

**Platforms:** UNIX only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you are running IBM DB2 on the AIX or Solaris platforms, verify that the Siebel database instance owner belongs to the primary group of the fenced user. If the instance owner is not part of this group, errors will occur during the Siebel database upgrade.

## Creating IBM DB2 Temporary Tablespaces and Bufferpools for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If your RDBMS is IBM DB2, verify that you have 16-KB and 32-KB temporary tablespaces to use for sorting and other SQL processing. Both the 16-KB and 32-KB temporary tablespaces require dedicated bufferpools.

### *To create a 16-KB temporary tablespace*

- 1 Create a 16-KB bufferpool with at least 5000 16-KB pages.

- 2 Create a 16-KB temporary tablespace as system managed space (SMS) that can be expanded to 2 GB of storage.

### *To create a 32-KB temporary tablespace*

- 1 Create a 32-KB bufferpool with at least 1000 32-KB pages.
- 2 Create a 32-KB temporary tablespace as SMS that can be expanded to 2 GB of storage.

## Analyzing IBM DB2 Custom Tablespace Requirements for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

There are four standard database managed tablespaces (DMS) that hold Siebel tables and indexes—a 4-KB, 16-KB, 32-KB tablespace, for tables, and a tablespace to hold indexes. The upgrade process moves tables between these spaces as required.

If you have placed Siebel tables in other tablespaces, the upgrade process will not move these tables if they grow to exceed the tablespace size during the upgrade. If one of these tables has an estimated page size after upgrade greater than its current page size, it will not fit in its tablespace after the upgrade, and the upgrade will fail.

Oracle provides a sizing utility that determines whether tables will increase in size to the point that they must be moved to a larger tablespace.

Run the utility before upgrading the database. If the sizing utility reports any problems, you must resolve them before you proceed with the upgrade.

### *To analyze tablespace requirements for DB2*

- 1 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 2 Type the following command line:

```
tblsize /U TABLEOWNER /P PASSWORD /C ODBC_DATASOURCE /F DDL_FILE  
/B DEFAULT_TABLESPACE /X DEFAULT_INDEXSPACE /K 16K_TABLESPACE /N 32K_TABLESPACE  
/O REPORT_FILENAME /L LOG_FILENAME
```

where:

- `TABLEOWNER` = Tableowner

- *PASSWORD* = Tableowner password
- *ODBC\_DATASOURCE* = Data source of the database
- *DDL\_FILE* = Absolute path to the DDL file (This file is called `ddl.ctl`, and it is located in the `dbsrvr/db2udb` directory.)
- *DEFAULT\_TABLESPACE* = Name of the 4-KB page standard Siebel tablespace
- *DEFAULT\_INDEXSPACE* = Name of the standard Siebel index space
- *16K\_TABLESPACE* = Name of the 16-KB page standard Siebel tablespace
- *32K\_TABLESPACE* = Name of the 32-KB page standard Siebel tablespace
- *REPORT\_FILENAME* = Name of the report generated by the utility
- *LOG\_FILENAME* = Name of the log file (The default name is `custtbl.log`.)

Example:

```
tblsize /U siebel /P siebel /C ssa /F d: /sea77/dbsrvr/db2udb/ddl.ctl /B siebel_4k  
/X siebel_idx /K siebel_16k /V siebel_32k /Q d: /sea77/dbsrvr/db2udb/report.txt /  
L $SIEBEL_ROOT/log/tblsize.log
```

- 3 Review the report generated by the utility to determine if the estimated table pagesize postupgrade is larger than the size of the actual 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
```

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

- 5 Move the tables from their old tablespaces to the new ones by running `ddl move`.

`ddl move` is a utility for moving tables from one tablespace to another tablespace. This utility is located in the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 6 To run `ddl move`, submit the following arguments:

```
ddl move /U TABLEOWNER /P TABLE_PASSWORD /C ODBC_DATASOURCE /E STOP_ON_DDL_ERROR  
/G GRANTEE /B TABLESPACE /X INDEX_TABLESPACE /M TABLE_NAME /L LOG_FILENAME  
/Z UCS2_DATABASE
```

where:

- *TABLEOWNER* = Tableowner of the database (Required)
- *TABLE\_PASSWORD* = Password of the tableowner of the database (Required)
- *ODBC\_DATASOURCE* = 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 tablespace that you are moving the table to
- *INDEX\_TABLESPACE* = Name of the index space that you are moving the table to
- *TABLE\_NAME* = Table Name Like Support (Default: N)
- *LOG\_FILENAME* = Name of the log file (The default name is ddlmove.log.)
- *UCS2\_DATABASE* = (Default: N)

## Verifying the DB2 Application Development Client for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The IBM DB2 Application Development Client must be installed on the RDBMS server. [Table 40](#) lists the required DB2 Application Development Client components.

Table 40. DB2 Application Development Client Components

Platform	DB2 Application Development Client Components
Windows	DB2 Application Development Client
AIX	Application Development Tools (ADT) ADT Sample Programs
HP	Application Development Tools for HP-UX
Solaris	Application Development Tools (ADT) ADT Sample Programs

For information on installing the Application Development Client, refer to the relevant IBM documentation.

# Identifying IBM DB2 Long Columns for Truncation in a Siebel Upgrade

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel 7.7, the maximum length for IBM DB2 long columns with a type of varchar was reduced to 16350 from 16383. The upgrade truncates long columns of type varchar that exceed 16,350.

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.

**CAUTION:** If you do not truncate the data in long varchar columns that exceed the maximum length specified below, a “data truncated” error will occur, and transaction processing and transaction routing may fail.

## *To identify and reduce the length of long varchar columns*

- 1 From any shell, open the script `chk16350.bat` (Windows) or `chk16350.ksh` (UNIX), 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_TBL0` = tableowner of the source database

`SRC_TBL0_PSWD` = tableowner 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` = The directory where you installed the Siebel Database Server files on the Siebel Server. For example, `C:\sea7xx\dbsrvr` (Windows). Edit the value “CHANGE\_ME”.

`SIEBEL_ROOT` = The directory where you installed the Siebel Server. For example, `C:\sea7xx\siebsrvr` (Windows). 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 16,350 characters.
- **update\_trunc.sql.** This SQL file generates update statements that truncate identified columns to 16,350 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.

## Installing IBM DB2 Stored Procedures and Functions for a Siebel Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The upgrade creates and binds the stored procedures.

User-defined functions (UDFs) for Siebel have changed at Siebel 7.x. Before upgrading the database, you must copy the new UDFs to the DB2 server. The upgrade process binds the UDF package to the DB2 installation.

Copying the new UDF package is a two-step process:

- Deleting the existing stored procedures and UDFs.
- Copying the stored procedure package to the DB2 server.

### Deleting the Existing Stored Procedures and UDFs

#### *To delete the existing stored procedures and UDFs*

- 1 On the DB2 server, navigate to the following directory in the DB2 installation:

Windows: sqllib\function

UNIX: sqllib/function

- 2 If the server is installed on a Windows platform, delete the following files:

nextseq.dll

nextseqd.exe

siebstat.dll

siebsdmy.dll

siebtrun.dll

siebfctst.dll

mappi ng. dll  
updt2col . dll  
updt\_addr\_org\_sp. dll  
updt\_addrorg\_i ntsp. dll  
updt\_addr\_per\_sp. dll

- 3 If the server is installed on a UNIX platform, delete the following files:

nextseq  
nextseqd  
siebstat  
siebug1  
siebsdmy  
mappi ng  
updt2col  
siebtrun  
siebfcst  
updt\_addr\_org\_sp  
updt\_addr\_per\_sp  
updt\_addrorg\_i ntsp

## Copying the Stored Procedure Package to the DB2 Server

### *To copy the stored procedure package to the DB2 server*

- 1 On the host where you installed the Siebel Database Server, navigate to the following directory:

Windows: *DBSRVR\_ROOT\db2udb\siebproc\server\_os*

UNIX: *DBSRVR\_ROOT/db2udb/siebproc/server\_os*

where:

*server\_os* is the operating system on which the DB2 server is installed.

- 2 Copy the file *siebproc* (*siebproc64* for 64 bit systems) to the *sqllib\function* (UNIX: *sqllib/function*) directory in the DB2 installation.



# 9

## Preparing an Oracle Database for a Siebel Upgrade

This chapter contains the following topics:

- [“Verifying Oracle Database Sort Order for a Siebel Upgrade” on page 169](#)
- [“Setting Oracle Database Configuration Parameters for a Siebel Upgrade” on page 170](#)
- [“Verifying Oracle Database Parameters for Multiple CPUs in a Siebel Upgrade” on page 172](#)
- [“Verifying the Oracle Database ODBC Definition for a Siebel Upgrade” on page 172](#)

### Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Verifying Oracle Database Sort Order for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The NLS\_SORT parameter determines the sort order of query returns. NLS\_SORT must be set to BINARY for the development environment upgrade, including the repository merge. It must also be set to BINARY when using Siebel Tools to compile an SRF file.

This setting is strongly recommended for production test environment upgrades and production upgrades.

### *To verify that your database is using binary sort order*

- 1 Run SQLPlus to connect to the Oracle database.
- 2 Issue the following query:  

```
SQL> SELECT * FROM NLS_DATABASE_PARAMETERS;
```
- 3 In the returned parameters, locate NLS\_SORT and verify that its value is BINARY.
  - If NLS\_SORT has a value of BINARY, then the sort order is binary and no action is required.
  - If NLS\_SORT is anything other than BINARY, reset the value to BINARY.

## Related Topics

[“About Database Sort Order in the Siebel Environment” on page 123](#)

# Setting Oracle Database Configuration Parameters for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This topic provides upgrade-specific settings for the Oracle RDBMS. Use the following strategy to set parameters:

- Read *System Requirements and Supported Platforms* on Siebel SupportWeb to determine the supported Oracle version and any required patches. If required, upgrade your Oracle database and client software to the supported version, using Oracle’s tools and documented procedures.
- Set parameters using the recommendations in Technical Note 582 and Alert 1011 on Siebel SupportWeb.
- Set additional parameters using the recommendations in *Siebel Installation Guide* for the operating system you are using. Recommendations are located in the chapter on configuring the RDBMS.
- Configure the database and set parameters as indicated in the following topics.
- After the upgrade, reset the configuration parameters to the values listed in *Siebel Installation Guide* for the operating system you are using and in Technical Note 582.

## General Configuration

Before upgrading an Oracle database, complete the following tasks:

- Verify that your Oracle clients and Oracle database server are at the same release level.
- **UNIX only.** If you have installed the Oracle 64-bit client on the Siebel Server, verify that \$ORACLE\_HOME/lib32 instead of \$ORACLE\_HOME/lib is included in LIBPATH (AIX), SHLB\_PATH (HP-UX), LD\_LIBRARY\_PATH (Solaris).
- Make sure tablespaces have enough free space to accommodate expected growth as estimated in [“About Siebel Database Growth Estimates” on page 128.](#)
- **pctincrease.** For upgrades, compute a high enough value for pctincrease on tablespaces that contain application tables and indexes so that upgrading does not create large numbers of extents.

- **pctfree.** Rebuild some of your larger tables with a large value for pctfree (30 or higher). Table size depends on which Siebel applications you have installed. For example, if you are upgrading Siebel Financial Services, S\_ASSET is a large table and S\_ADDR\_ORG is not used at all.

You must increase pctfree before the upgrade because many new columns are added to tables during the upgrade. Migrating data into the new columns during the upgrade is likely to cause row chaining, which degrades upgrade performance.

Examples of large tables for upgrades from Siebel 6.x are:

- s\_contact
  - s\_evt\_act
  - s\_srv\_req
  - s\_src
  - s\_org\_ext
  - s\_addr\_org
- **DB\_CACHE\_SIZE.** Set this parameter to a minimum of 394264576.
  - **SORT\_AREA\_SIZE.** Set this parameter to a minimum of 1524288. This significantly reduces the time required to complete a repository merge.
  - Other ini t. ora parameters that the DBA should review are:
    - sort\_area\_retained size
    - log\_buffers
  - **UNDO\_MANAGEMENT (Oracle 9i and later).** Set UNDO\_MANAGEMENT = MANUAL before the repository merge. This turns off Automatic Undo Management (AUM). You can turn AUM back on after the repository merge, as desired. For more information on how AUM affects upgrade, see Alert 848 on Siebel SupportWeb.
  - **Rollback Segments.** Verify that you have only one large rollback segment on line that is appropriately sized so that the largest of transactions can be accommodated. Take all other rollback segments off line.

The upgrade may affect some of the largest tables in your implementation of Siebel 7.x, causing them to grow by as much as 40%.

Customer experience has shown that repository merges involving multiple languages can require a rollback segment as large as 1 GB.

## Query Optimizer Settings

For the upgrade, use the query optimizer settings listed in Technical Note 582 and Alert 1011 on Siebel SupportWeb.

## Verifying Oracle Database Parameters for Multiple CPUs in a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you have multiple CPUs, verify that the following parameters are set correctly. For information on settings, refer to Oracle documentation:

- `parallel_max_servers`. **To enable use of multiple CPUs, this must be set to a number greater than 1.**
- `parallel_min_servers`
- `db_block_size` (Oracle 8i)
- `_db_block_size` (Oracle 9i)

## Verifying the Oracle Database ODBC Definition for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

When you install a Siebel Server, the installer creates the ODBC definition for the Oracle database you specify.

An ODBC definition for connecting to an Oracle 8i database cannot be used by Siebel Applications for connecting to an Oracle 9i database. An Oracle 9i ODBC definition cannot be used to connect to an Oracle 8i database.

When you upgrade your production test environment and your production environment, you must manually create an ODBC definition for connecting to your development environment database. If the development environment database is Oracle 9i, you must edit the ODBC definition after it has been created.

### To verify or edit an ODBC definition under Windows

- 1 Start the registry editor (regedit).

**CAUTION:** Editing the registry can adversely affect the operating system. Be sure you understand how to use regedit correctly.

- 2 Navigate to the following location:

HKEY\_LOCAL\_MACHINE\Software\ODBC\ODBC.INI\*ODBC\_Name*

Where:

*ODBC\_Name* is the ODBC name for the Oracle database.

- 3 In the data display pane, verify that the following entries are present:

Name	Type	Data
ColumnsAsChar	REG_SZ	1
ColumnSizeAsCharacter	REG_SZ	1

If these entries are not present, right-click in the data display pane and choose New > String Value to add them.

- 4 Step off the *ODBC\_Name* and return to it. Verify that the two new entries are present and correct.
- 5 Close the Registry Editor and reboot.

### To verify or edit an ODBC definition under UNIX

- 1 Navigate to the following file in the Siebel Server installation directory:

\$SIEBEL\_ROOT/sys/.odbc.ini

- 2 Open the .odbc.ini file and add the following two entries:

ColumnsAsChar=1

ColumnSizeAsCharacter=1

- 3 Save the file.
- 4 Stop and restart any processes that are using this .odbc.ini file.



# 10 Preparing an Microsoft SQL Server Database for a Siebel Upgrade

This chapter contains the following topics:

- [“Verifying Microsoft SQL Server Sort Order for a Siebel Upgrade”](#) on page 175
- [“Setting Microsoft SQL Server Temporary Space Size for a Siebel Upgrade”](#) on page 176
- [“Setting Microsoft SQL Server Configuration Parameters for a Siebel Upgrade”](#) on page 177
- [“Rebuilding Microsoft SQL Server Clustered Indexes for a Siebel Upgrade”](#) on page 178

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Verifying Microsoft SQL Server Sort Order for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** MS SQL Server only.

**Platforms:** MS Windows only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Verify that the sort order of the master database and the database instance are the same. This prevents repository merge failure due to sort order mismatch.

Binary sort order is required for the development environment upgrade and strongly recommended for the production environment upgrades.

When you install Microsoft SQL Server, the collation method of the database instance is set by default to dictionary sort order. Every database you create thereafter inherits this setting.

When you create a database, you can accept the inherited sort order or specify the sort order. It is recommended that you set the sort order to binary at the Microsoft SQL Server instance level so that this sort order is inherited by newly created databases.

The sort order of the master database cannot be changed without rebuilding the instance. Consult your Microsoft documentation for instructions on setting database collation.

### *To verify that your database was created using a binary collation sequence*

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

```
sp_hel psort
```

This command provides a sort order description.

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

```
Lati n1_General _BIN
```

If you find that your Microsoft SQL Server database was not created using a binary collation sequence, you must rebuild your database and reload your data. Review the Microsoft documentation for detailed instructions.

### **Related Topics**

[“About Database Sort Order in the Siebel Environment” on page 123.](#)

# Setting Microsoft SQL Server Temporary Space Size for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** MS SQL Server only.

**Platforms:** MS Windows only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Set the size of the database that Microsoft SQL Server uses for temporary space needed to execute queries.

### *To setup TEMPDB space*

- 1 Make TEMPDB as big as the biggest table in the Siebel database, or half the size of the Siebel database.

- 2 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 upgrade activity. Alternatively, you can set MAXSIZE to the size of the biggest table or to 50% of the size of the Siebel database.

- 3 Consider putting TEMPDB on a separate drive to improve performance.

- 4 Execute dbcc shri nkdatabase against TEMPDB.

# Setting Microsoft SQL Server Configuration Parameters for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** MS SQL Server only.

**Platforms:** MS Windows only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This topic provides upgrade-specific settings for the Microsoft SQL RDBMS. Use the following strategy to set parameters:

- Set parameters using the recommendations in *Siebel Installation Guide* for the operating system you are using. Recommendations are located in the chapter on configuring the RDBMS.
- For the upgrade, revise the configuration parameters listed below.
- After the upgrade, reset the configuration parameters to the values listed in *Siebel Installation Guide* for the operating system you are using.

[Table 41](#) lists upgrade settings for Microsoft SQL Server database parameters. For parameters not listed in this table, it is recommended that you accept the default settings.

Most of the parameter settings in the table are the default settings.

Table 41. Microsoft SQL Configuration Parameters

Parameter	Setting/Comment
Max. degree of parallelism	1
Cost threshold for parallelism	5
Fill factor (%)	90
Index create memory (KB)	0

For the Siebel database, set the following options to ON (enabled) for the upgrade:

- **truncate log on chkpt.** Set this option to ON (enabled) for upgrade only. Also, for upgrade only, execute the `alter` command against the Siebel database, specifying `set recovery simple`.
- **torn page detection.**
- **auto create statistics.**
- **auto update statistics.**
- **Database size.** 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.

- For a full list of recommended settings for your postupgrade production environment, see the chapter on configuring the RDBMS in *Siebel Installation Guide* for the operating system you are using.

## Rebuilding Microsoft SQL Server Clustered Indexes for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** MS SQL Server only.

**Platforms:** MS Windows only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

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 Microsoft SQL Server `create index` command with `drop_existing` clause to rebuild large tables with high fillfactor (60%-70%).

# 11 Preparing Siebel Application Data for Upgrade

This chapter contains the following topics:

- [“Preparing Siebel Workflow Processes for Upgrade” on page 179](#)
- [“Preparing Siebel Customized Seed Data for Upgrade” on page 181](#)
- [“Migrating Siebel Household Data” on page 182](#)
- [“Setting Up Campaign Status Values for Siebel Marketing” on page 184](#)
- [“Preserving Siebel Marketing Segment Descriptions” on page 187](#)
- [“Preparing Siebel Mobile User Data for Upgrade” on page 187](#)
- [“Setting the Value of Siebel S\\_SRC\\_PAYMENT.TYPE\\_CD for Upgrade” on page 188](#)
- [“Preparing Siebel Address Data for Upgrade” on page 189](#)
- [“Migrating Siebel Address Data from Custom Extension Columns” on page 190](#)
- [“Preparing Siebel Configurator Data for Upgrade” on page 197](#)
- [“Resolving Duplicate Row IDs for a Siebel Upgrade” on page 198](#)
- [“Preparing Siebel eChannel Data for Upgrade” on page 199](#)
- [“Preparing Siebel Forecasting Data for Upgrade” on page 199](#)
- [“Preparing Siebel Products and Quotes Data for Upgrade” on page 200](#)
- [“Preparing Siebel Financial Services Tables for Upgrade” on page 201](#)
- [“Preparing Siebel Financial Services Tables for Upgrade” on page 201](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Preparing Siebel Workflow Processes for Upgrade

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This task prepares workflows for upgrade.

At Siebel 7.7 the following changes were made to workflow management:

- The upgrade migrates custom workflows from S\_WF\_\* tables to S\_WFR\_\* tables in the Siebel Repository. The upgrade also updates seed workflows.
- The tables S\_WF\_PROP\_VAL and S\_WF\_STEP\_INST no longer store workflow process instances. Workflow process instances are now stored in S\_WFA\_INST\_LOG, S\_WFA\_INSTP\_Log, and S\_WFA\_STPRP\_LOG.

## Upgrades from Releases Prior to Siebel 7.7

Perform the following task before upgrading from releases prior to Siebel 7.7.x.

### *Preparing workflow processes for upgrade*

- 1 Development environment.** Verify that all customized workflows and all seed workflows from the production database are present in the development environment Siebel database.

If the workflows from the production environment are not present, they will not be moved to the Siebel Tools Repository. You will have to manually move them using database tools.

Oracle provides a script and an input file for migrating workflow data between your production database and your development database. See Alert 1016 on Siebel SupportWeb.

- 2 All environments.** Verify there are no queued or running workflows.
- 3 All environments.** Purge S\_WF\_PROP\_VAL and S\_WF\_STEP\_INST.

For information on purging these tables, see the topic on purging workflow process instances in *Siebel Business Process Designer Administration Guide*.

## Upgrades from All Releases

Perform the following task for all upgrades

### *Preparing workflow processes for upgrade*

- 1** Deactivate all workflow processes.  
This ensures that run-time event actions associated with old workflows are cleaned up properly.
- 2** Verify that all tasks with status = QUEUED are removed from S\_SRM\_REQUEST.  
The upgrade process does not clear these.

### Related Topics

[“Upgrade Planning for Siebel Workflow Designer” on page 147](#)

# Preparing Siebel Customized Seed Data for Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

Modified seed data are not upgraded, even if the upgraded seed data are required for an application to run normally, with the following exceptions:

- If you modify seed data through the GUI or through EIM, the repository merge preserves the modifications using the same logic as for other repository objects.

Before starting the upgrade, evaluate whether you need to retain your seed data modifications. If practical, consider discarding the modifications. After the upgrade, you can reimplement those modifications that do not interfere with operation of applications.

Seed data are records provided in the Siebel database tables as part of a Release. Seed data provides information needed to run a Siebel application. Examples of seed data are the values in a List of Values (LOV) definition, default mappings of views to responsibilities, and predefined queries.

The ROW\_ID value for seed data records provided in a Release begins with 0 (zero). In addition, these records have a default LAST\_UPD value of 1980\_01\_01. The LAST\_UPD field stores the date when the record was last updated.

If you modify a seed data record, the value of LAST\_UPD is changed from 1980\_01\_01 to the date the modification was made. If you add a new seed data record, the ROW\_ID does not begin with 0, and the LAST\_UPD value is the record creation date.

When you perform the upgrep part of the upgrade, the dataimp utility upgrades seed data records as follows:

- a For records where ROW\_ID begins with 0, it erases seed data records where the value for LAST\_UPD is 1980\_01\_01, unless prevented by scripting.
- b Dataimp replaces these records with those contained in seed data files included in the Release. In the new records, the value for LAST\_UPD is 1980\_01\_01
- c Dataimp does not erase and replace records where the value for LAST\_UPD is later than 1980\_01\_01. Instead, for each of these records, dataimp writes an error to the log file. The error is benign and does not cause the upgrade to fail.

Use these error entries to verify which seed data records were not changed.

For seed data records you have modified, you can discard the modifications by changing the value of LAST\_UPD to 1980-01-01. This causes dataimp to replace these records with those from the new Release. For seed data records you have created, the upgrade process retains these records.

## *To discard seed data modifications*

- 1 In Siebel Tools, select the Table object.

- 2 In the list applet, query for "\*" in the Seed Filter column.  
The query returns a list of tables containing seed data.
- 3 Use one of the following scripts to set LAST\_UPD to 1980-01-01 for customized seed data records in these tables. In the scripts, *tablename* is the name of table containing seed data.

Database	Script
Oracle	UPDATE <i>tablename</i> SET LAST_UPD = TO_DATE(' 1980-01-01' , ' YYYY-MM-DD' ) WHERE ROW_ID LIKE ' 0%' AND LAST_UPD > TO_DATE(' 1980-01-01' , ' YYYY-MM-DD' )
IBM DB2UDB and IBM DB2 for z/OS	UPDATE <i>tablename</i> SET LAST_UPD = TIMESTAMP(' 1980-01-01-00.00.00' ) WHERE ROW_ID LIKE ' 0%' AND LAST_UPD > TIMESTAMP(' 1980-01-01-00.00.00' )
Microsoft MS SQL	UPDATE <i>tablename</i> SET LAST_UPD = CONVERT(DATETIME, ' 1980-01-01' ) WHERE ROW_ID LIKE ' 0%' AND LAST_UPD > CONVERT(DATETIME, ' 1980-01-01' )

**Related Topics**

- ["About the Siebel Database Upgrade Log Files" on page 80](#)
- ["About the Siebel Repository Merge" on page 89](#)
- ["Upgrade Planning for Siebel Workflow Designer" on page 147](#)

## Migrating Siebel Household Data

**Upgrades:** Applies to Siebel Financial Services upgrades from Siebel 6.x. Applies to Siebel Financial Services upgrades from 7.x that have retained the Siebel 6.x form of household associations.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Siebel 7.0.x introduced the Party model. This changed the way relationships between households and entities, such as activity and claim, are handled.

You have two options for migrating household data:

- Migrating household relationships to the Party model (recommended)
- Retaining the Siebel 6.x form of household relationships

### Migrating Household Relationships to the Party Model

To check household data integrity and support migration of household data to the Party model, you must run the household verification script (HH\_MIG\_populate.sql).

The script verifies that at least the same number of entities will belong to a household after the upgrade as belong to it before the upgrade.

The household verification script makes the following assumptions:

- A household has at least one contact.
- The primary contact of a Policy/Financial Account is one of the contacts associated with this Policy/Financial Account.
- The primary contact of a Claim is one of the contacts associated with this Claim.
- The primary contact of an Opportunity is one of the contacts associated with this Opportunity.
- The primary contact of a Company is one of the contacts associated with this Company.

The script populates a temporary table with data, TEMP\_HH\_OBJ\_MIG and generates a report based on an output file. Output is in the form of row IDs. The script verifies that every household associated with an entity includes a contact associated with that entity.

If there is no output, this means data integrity is good, and no action is required. If you receive output, you must examine the relationship between contacts and households.

Do not run the household verification script if you intend to maintain the Siebel 6.x method of handling household relationships.

### To run the household verification script

1 Type the following command:

- Windows:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /a /c REM /separator / /O
OutputFileLocation\HH_Mig_populate.txt
/L LogFileLocation\HH_Mig_populate.log ScriptLocation\HH_Mig_populate.sql /v y
```

- UNIX:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /a /c REM /separator / /O
OutputFileLocation/HH_Mig_populate.txt
/L LogFileLocation/HH_Mig_populate.log ScriptLocation/HH_Mig_populate.sql /v
```

where:

Tableowner = Tableowner

Password = Tableowner password

ODBCDataSource = Data source of the database

OutputFileLocation = Location of the output file:

- Windows: *SIEBEL\_ROOT\log\HH\_Mig\_populate.txt*
- UNIX: *\$SIEBEL\_ROOT/log/HH\_Mig\_populate.txt*

LogFileLocation = Location of the log file :

- Windows: *SIEBEL\_ROOT\log\HH\_Mig\_populate.log*

- ❏ UNIX: `$SIEBEL_ROOT/Log/HH_Mi g_populate.log`  
ScriptLocation = Location of the script:
- ❏ Windows: `DBSRVR_ROOT\database_platform\HH_Mi g_populate.sql`
- ❏ UNIX: `DBSRVR_ROOT/database_platform/HH_Mi g_populate.sql`

Windows example:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /a /c REM /separator / /O  
C: \sea7xx\si ebsrvr\Log\HH_Mi g_populate.txt /L  
C: \sea7xx\si ebsrvr\Log\HH_Mi g_populate.log  
C: \sea7xx\dbsrvr\DB2UDB\HH_Mi g_populate.sql /v y
```

- 2 If you receive output, review the temporary table and check the following for each contact. Make corrections as needed:
  - Contact is correct and household is incorrect.
  - Contact is incorrect and household is correct.
  - Contact is incorrect and household is incorrect.

## Retaining the Siebel 6.x Form of Household Relationships

You have the option to retain the Siebel 6.x style of handling household relationships. However, this is not recommended. If you choose not to upgrade to the new household design, do the following:

- Verify that the temporary table TEMP\_HH\_OBJ\_MIG does not exist in the database.
- Do not run the HH\_Mi g\_populate.sql script.
- Before doing the upgrade, contact Siebel Technical Support or Oracle's Siebel Expert Services and do the following:
  - Verify that the business components needed to support the Siebel 6.x form of household relationships are present in the new release.
  - Obtain instructions for revising these business components to support the Siebel 6.x form of household relationships.

### Related Topics

["About the Siebel Party Model" on page 124.](#)

# Setting Up Campaign Status Values for Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This optional step is not required for upgrade processing, but will simplify program and campaign reporting by making historical values equivalent with the values used as of Siebel 7.7.

The database upgrade process expects the standard Status values for the following applets. The process uses the standard values to convert records to the new campaign model during the upgrade:

- Campaigns (Business Component is Campaign)
- Campaign Plans (Business Component is DBM Campaign)
- Campaign Occurrences (Business Component is Campaign Occurrences)

If you customized the values for the Status field for these objects, convert the Status (S\_SRC.STATUS\_CD) of any existing records to the standard values before doing the upgrade.

Any campaign records that do not have standard Status values will not create Load Wave records during the upgrade and will not be displayed as launched campaigns in the Campaign Agent views.

Optionally, you can also migrate the status values for the following components:

- Program (Business Component is Program Container)
- Stage (Business Component is Program (DBM))

[Table 42](#) lists the standard Status values for each object.

Table 42. Standard Status Values

Object	Standard Status Values
Program Plan	Planned Active Completed
Stage Plan	Planned Active Completed
Campaign Plan	Planned Active Completed

Table 42. Standard Status Values

Object	Standard Status Values
Campaign	Planned Active Completed
Campaign Occurrence	Pending Active Finished Manual Error Cancelled

Table 43 lists the search specifications for locating records containing Status values. Siebel 7.5 terminology is used for the object names.

Table 43. Search Specifications for Locating Records

Object	Business Component	Base Table	Search Specification
Program Plan	Program Container	S_SRC	WHERE SUB_TYPE = 'MARKETING_CAMPAIGN' AND CAMP_TYPE_CD = 'PROGRAM_CONTAINER'
Stage Plan	Program (DBM)	S_SRC	WHERE SUB_TYPE = 'MARKETING_CAMPAIGN' AND CAMP_TYPE_CD = 'PROGRAM'
Campaigns	Campaign	S_SRC	WHERE SUB_TYPE = 'MARKETING_CAMPAIGN' AND CAMP_TYPE_CD IS NULL AND CUST_TRGT_METH_CD IS NULL AND TMPL_ID IS NULL
Campaign Plans	DBM Campaign	S_SRC	WHERE SUB_TYPE = 'MARKETING_CAMPAIGN' AND CAMP_TYPE_CD IS NULL AND CUST_TRGT_METH_CD IS NOT NULL
Campaigns (Occurrences)	Campaign Occurrences	S_SRC	WHERE SUB_TYPE = 'MARKETING_CAMPAIGN' AND CAMP_TYPE_CD IS NULL AND CUST_TRGT_METH_CD IS NULL

### Related Topics

["Upgrade Planning for Siebel Marketing" on page 141](#)

# Preserving Siebel Marketing Segment Descriptions

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

When upgrading from a prior release of Siebel Marketing to Siebel 7.7 or later, the database upgrade overwrites the Description field for all Segment records (S\_CALL\_LST.DESC\_TEXT) with the string “DO NOT USE: Segment from previous release.” If you want to preserve segment descriptions, migrate the existing descriptions to another text extension column before running the upgrade.

## Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Preparing Siebel Mobile User Data for Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic applies primarily to developers running the Mobile Web Client in the development environment and to end users in the production environment. This topic applies to the production test environment only if it has Mobile Web Client users.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

After synchronizing, mobile users must make no further changes to their local databases until the upgrade has been completed. Any changes made during the upgrade are lost when they are reinitialized following the upgrade.

Complete the following steps before beginning the upgrade of either a development environment or a production environment. For additional information on these steps, see *Siebel Remote and Replication Manager Administration Guide* and *Siebel System Administration Guide*.

## *To prepare mobile users for the database upgrade*

- 1 Perform a partial synchronization for mobile users, sending all transactions to the server database.
- 2 Verify that Mobile Web Clients have synchronized and that all changes have been merged into the server database as follows:

- 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:  
Windows: `SI EBEL_ROOT\docki ng\MOBI LEUSERNAME.`  
UNIX: `$SI EBEL_ROOT/docki ng/MOBI LEUSERNAME.`  
Transaction files are in the format *number.dx*; for example, 00000023. dx.
- b** Check the mobile users Remote Status view and resolve any insert conflicts.
- c** Log onto a Siebel Business 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.
- d** 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 Web Clients with the database server, stop or disable all Siebel Remote components on all Siebel Servers.
- 4** Disconnect all Web Clients from the Siebel Server by stopping the appropriate Application Object Managers, as described in *Siebel System Administration Guide*.
- 5 Upgrades from Siebel 7.5x and earlier.** Make sure that Dedicated Web Clients have disconnected from the Siebel Database Server.

The method you use to do this depends 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.

## Setting the Value of Siebel S\_SRC\_PAYMENT.TYPE\_CD for Upgrade

**Upgrades from:** Siebel 6.x & 7.0.x.

**All environments.** This topic applies to all Siebel environments.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

At Siebel 7.5, S\_SRC\_PAYMENT.TYPE\_CD became a required (NOT NULL) column. If TYPE\_CD is null, the value is set to Payment during the upgrade.

Payment indicates 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.

Before the upgrade, review records where TYPE\_CD is null. For those where you do not want the upgrade to set the value of TYPE\_CD to Payment, set the value of TYPE\_CD.

### To set the value of *TYPE\_CD*

- 1 In S\_SRC\_PAYMENT, query for records where TYPE\_CD is null.
- 2 For the desired records, use Siebel EIM to set a value for TYPE\_CD.  
Use the values in the field's LOV.

## Preparing Siebel Address Data for Upgrade

### Upgrades:

- From Siebel Industry Applications (SIA) 6.x, (except Communications, Media, and Energy) to Siebel SIA 7.7.x
- From Siebel Financial Services 6.x and 7.0.x to Siebel SIA 7.7.x
- From Siebel Business Applications 6.x and 7.x to Siebel Business Applications 7.7.x as part of upgrading to Siebel SIA 7.7.x
- From Siebel Financial Services 6.2.1 on IBM z/OS platforms to Siebel SIA 7.7.x on IBM z/OS platforms

**NOTE:** This topic does not apply to Siebel Business Applications (HOR) that you are upgrading to a later release of Siebel Business Applications (HOR).

**Environments:** Production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

At Siebel 7.7, the way address data is stored is changed. To prepare for the revised storage scheme, you must verify that there are no records with the same row IDs within or across the tables S\_ADDR\_PER and S\_ADDR\_ORG.

**CAUTION:** There must be no duplicate row IDs in these tables or the upgrade will fail.

### To prepare address data for upgrade

- 1 Run rpt\_dup\_addr\_rowi ds. sql against the Siebel database. The script is located in the following directory:

Windows: *DBSRVR\_ROOT\database\_platform*

UNIX: *DBSRVR\_ROOT/database\_platform*

where

*database\_platform* is the database type, for example DB2.

- 2 Review the output generated by the script.
- 3 If the output contains records with duplicate row IDs, use EIM or the application to delete unwanted records.

- 4 After addressing all the duplicate row IDs, rerun the script and verify there are no more duplicates.

### Related Topics

[“Upgrade Planning for Migrating Siebel Address Data” on page 139](#)

[“Migrating Siebel Address Data from Custom Extension Columns” on page 190](#)

# Migrating Siebel Address Data from Custom Extension Columns

## Upgrades:

- From Siebel Industry Applications (SIA) 6.x, (except Communications, Media, and Energy) to Siebel SIA 7.7.x
- From Siebel Financial Services 6.x and 7.0.x to Siebel SIA 7.7.x
- From Siebel Business Applications 6.x and 7.x to Siebel Business Applications 7.7.x as part of upgrading to Siebel SIA 7.7.x
- From Siebel Financial Services 6.2.1 on IBM z/OS platforms to Siebel SIA 7.7.x on IBM z/OS platforms

**Environments:** Development and production test.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**NOTE:** This topic does not apply to Siebel Business Applications (HOR) that you are upgrading to a later release of Siebel Business Applications (HOR).

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you have custom extension columns on the Address table S\_ADDR\_ORG, you must manually edit your upgrade scripts to prevent loss of data during an upgrade. Perform the following task, and contact Siebel Technical Support or Oracle’s Siebel Expert Services if you need assistance.

**Prerequisite.** You must have run the Database Server Configuration Utilities in upgrep mode. This creates the files required for this procedure. You perform this procedure before running the Upgrade Wizard.

## *To migrate address data from custom extension columns*

- 1 **Development environment upgrades only.** Open the ddl.ctl file with a text editor.

Here is the path to the file in the Siebel Database Server installation:

Windows: *DBSRVR\_ROOT\database\_platform*

UNIX: *DBSRVR\_ROOT/database\_platform*

where

*database\_platform* is the database type, for example DB2.

- 2 Development environment upgrades only.** Add the custom extension columns from S\_ADDR\_PER and S\_ADDR\_ORG that are attributes of the association of the address to contact or account to the S\_CON\_ADDR definition.

Example of the file:

```
[Object nnn]
```

```
Type = Table
```

```
Name = S_CON_ADDR
```

```
Space = USERSPACE1
```

```
Column 1 = ROW_ID VARCHAR(15) OTNULL
```

```
Column 2 = CREATED TIMESTAMP NOTNULL DEFAULT %NOW%
```

```
Column 3 = CREATED_BY VARCHAR(15) NOTNULL
```

```
Column 4 = LAST_UPD TIMESTAMP NOTNULL DEFAULT %NOW%
```

```
Column 5 = LAST_UPD_BY VARCHAR(15) NOTNULL
```

```
Column 6 = DCKING_NUM NUMERIC(22, 7) DEFAULT 0
```

```
... ..
```

```
Column 46 = RURAL_ROUTE_NUM VARCHAR(10)
```

```
Column 47 = START_DT TIMESTAMP
```

```
Column 48 = X_CUSTOM_EXTENSION_COLUMN1 VARCHAR(15)
```

```
Column 49 = X_CUSTOM_EXTENSION_COLUMN2 VARCHAR(15)
```

- 3 Development environment upgrades only.** In the ddl.ctl file, add the custom extension columns from S\_ADDR\_ORG that are attributes of the address to the S\_ADDR\_PER definition (provided corresponding columns do not already exist on S\_ADDR\_PER) similar to the example illustrated in the previous step.
- 4 Development environment and production test environment upgrades.** In the file preschm\_sia.sql, add the custom extension columns for S\_CON\_ADDR to the migration scripts if any custom extension column was added to the definition of S\_CON\_ADDR in the ddl.ctl file in the preceding steps.

Here is the path to the file in the Siebel Database Server installation:

Windows: *DBSRVR\_ROOT\database\_platform\upgrade\version*

UNIX: *DBSRVR\_ROOT/database\_platform/upgrade/version*

Example of the script (this example is for inserts into S\_CON\_ADDR from S\_ADDR\_PER. (The example also applies to custom extension columns to be inserted on S\_CON\_ADDR from S\_ADDR\_ORG):

```
Insert into S_CON_ADDR
```

```
(ROW_ID
, CREATED
, CREATED_BY
, LAST_UPD
, LAST_UPD_BY
, MODIFICATION_NUM
, CONFLICT_ID
, ADDR_PER_ID
, ADDR_TYPE_CD
, CONTACT_ID
, EMAIL_ADDR
, FAX_PH_NUM
, PH_NUM
, ADDR_MAIL_CD
, END_DT
, START_DT
, ACTIVE_FLG
, X_CUSTOM_EXTENSION_COLUMN1
, X_CUSTOM_EXTENSION_COLUMN2)
select
ROW_ID
, CREATED
, CREATED_BY
, LAST_UPD
, LAST_UPD_BY
, MODIFICATION_NUM
, CONFLICT_ID
, ROW_ID
, ADDR_TYPE_CD
, PER_ID
```

```

, EMAIL_ADDR
, FAX_PH_NUM
, PH_NUM
, ADDR_MAIL_CD
, END_DT
, START_DT
, case
when ACTIVE_FLG is null then 'Y'
else ACTIVE_FLG
end
, X_CUSTOM_EXTENSION_COLUMN1
, X_CUSTOM_EXTENSION_COLUMN2
from S_ADDR_PER APT
where not exists
(select 'x' from S_CON_ADDR CAD
where CAD.CONTACT_ID = APT.PER_ID
and CAD.ADDR_PER_ID = APT.ROW_ID
and CAD.CONFLICT_ID = APT.CONFLICT_ID
)
and not exists
(select 'x' from S_CON_ADDR CAD1
where CAD1.ROW_ID = APT.ROW_ID
)
and APT.PER_ID is not null

```

- 5 Development environment and production test environment upgrades.** In the file `preschm_sia.sql`, add the custom extension columns for `S_ADDR_PER` to the migration script if any custom extension column was added to the definition of `S_ADDR_PER` in the `ddl.ctf` file.

Here is the path to the file in the Siebel Database Server installation:

Windows: `DBSRVR_ROOT\database_platform\upgrade\version`

UNIX: `DBSRVR_ROOT/database_platform/upgrade/version`

Example of the file:

```
insert into S_ADDR_PER
(ROW_ID
, CREATED
, CREATED_BY
, LAST_UPD
, LAST_UPD_BY
, MODIFICATION_NUM
, CONFLICT_ID
, DISA_CLEANSER_FLG
, ADDR
, ADDR_NUM
, ADDR_TYPE_CD
, CITY
, COMMENTS
, COUNTRY
, COUNTY
, INTEGRATION_ID
, INTEGRATION2_ID
, INTEGRATION3_ID
, PROVINCE
, STATE
, ZIPCODE
, ADDR_NAME
, NAME_LOCK_FLG
, DESCRIPTOR
, EMAIL_ADDR
, FAX_PH_NUM
, PH_NUM
, ADDR_LINE_2
, ADDR_LINE_3
```

```
, DCKING_NUM
, LATITUDE
, LONGITUDE
, ACTIVE_FLG
, ADDR_MAIL_CD
, ADDR_SUB_CD
, END_DT
, RURAL_ROUTE_NUM
, START_DT
, X_CUSTOM_EXTENSION_COLUMN1
, X_CUSTOM_EXTENSION_COLUMN2
)
select
ROW_ID
, CREATED
, CREATED_BY
, LAST_UPD
, LAST_UPD_BY
, MODIFICATION_NUM
, CONFLICT_ID
, DISA_CLEANSER_FLG
, ADDR
, ADDR_NUM
, ADDR_TYPE_CD
, CITY
, COMMENTS
, COUNTRY
, COUNTY
, INTEGRATION_ID
, INTEGRATION2_ID
```

```
, INTEGRATION3_ID
, PROVINCE
, STATE
, ZIPCODE
, case when ADDR_NAME = (select t.DUP_ADDR_NAME
    from    TMPTBL_ADDR t
    where t.DUP_ADDR_NAME = S_ADDR_ORG.ADDR_NAME and t.MIN_ROW_ID <>
    S_ADDR_ORG.ROW_ID
        )
    then {fn concat({fn concat({fn rtrim({fn substring(ADDR_NAME, 1, 99- {fn
length(ROW_ID)}})}), ' : ')}, ROW_ID)}
    else ADDR_NAME
end
, NAME_LOCK_FLG
, DESCRIPTOR
, EMAIL_ADDR
, FAX_PH_NUM
, PH_NUM
, ADDR_LINE_2
, ADDR_LINE_3
, DCKING_NUM
, LATITUDE
, LONGITUDE
, ACTIVE_FLG
, ADDR_MAIL_CD
, ADDR_SUB_CD
, END_DT
, RURAL_ROUTE_NUM
, START_DT
, X_CUSTOM_EXTENSION_COLUMN1
, X_CUSTOM_EXTENSION_COLUMN2
```

from S\_ADDR\_ORG

- 6 Development environment upgrades only.** In Siebel Tools, add the custom extension columns defined in the previous steps to S\_CON\_ADDR and S\_ADDR\_PER to the Prior Customer Repository.

Remember that the Prior Customer Repository is the renamed Siebel Repository for the release you are upgrading from.

- 7 Development environment upgrades only.** Configure your application to expose the custom extension columns added to S\_ADDR\_PER.

#### Related Topics

[“Upgrade Planning for Migrating Siebel Address Data” on page 139](#)

[“Preparing Siebel Address Data for Upgrade” on page 189](#)

## Preparing Siebel Customizable Product Data for Upgrade

**Upgrades from:** All Siebel releases. Exception: does not apply to upgrades from Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

### Customizable Products in Work Spaces

The upgrade does not migrate unreleased customizable products in work spaces. If you want to migrate unreleased customizable products, you must release them before the upgrade. This includes products with components and products with attributes.

### Class Products

Verify that the Orderable flag is not set for class products. When this flag is not set, class products do not display as selectable products in quotes and orders after the upgrade.

The upgrade converts class products to a product and a product class. The upgrade sets the Product Class property for the product to Product Class.

#### Related Topics

[“Upgrade Planning for Siebel Configurator” on page 149](#)

## Preparing Siebel Configurator Data for Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Siebel Configurator’s data model migration is embedded in the Siebel standard upgrade process. However, in order to make sure that the model is upgrade properly, customers must have appropriate model product associations. For more information on this and other recommended pre-upgrade tasks, see Technical Note 428 on Oracle’s Siebel SupportWeb.

## Resolving Duplicate Row IDs for a Siebel Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development, production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

During the upgrade from Siebel 6.x to 7.7.x, ROW\_IDs from S\_EMPLOYEE, S\_CONTACT, S\_ORG\_INT, S\_ORG\_EXT, S\_POSTN, and S\_ORG\_GROUP are inserted into S\_PARTY.ROW\_ID. Although row IDs are typically unique across the entire schema, there may be rare instances when row IDs are shared across these source tables.

If duplicate row IDs are not resolved before the upgrade, the upgraded S\_PARTY data will be defective because there will be mismatches between the base S\_PARTY record and the corresponding extension table record.

After you install the Siebel Database Server software, but before you upgrade, you must identify and resolve any duplicate row IDs in your Siebel Database.

### *To resolve duplicate row IDs across source tables*

- 1 Run the script, `Find_DUP_S_PARTY_ROW_IDs.sql`, located in the following directory:

WINDOWS: `DBSRVR_ROOT\DATABASE_PLATFORM`

UNIX: `DBSRVR_ROOT/DATABASE_PLATFORM`

This SQL script generates a list of duplicate row IDs (if any) and the tables containing the rows.

- 2 Choose one of the records that has a duplicate row ID.

Choose the record with the smallest number of references to other records.

- 3 Copy the record.

This creates a new record with a unique row ID.

- 4 Use the Merge Records command to merge the original record with the copied record.

Records that were associated with the original record are now associated with the copied record.

## Preparing Siebel eChannel Data for Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

For upgrades from Siebel 6.x to 7.7.x, if you have modeled your partners as both Accounts and Divisions (or Organizations), merge these records so that there is only one record for each partner company.

Similarly, if you have modeled partner employees as both Contacts and Employees, merge these records so that there is one record for each person.

For assistance with merging records, contact Oracle's Siebel Expert Services.

## Preparing Siebel Forecasting Data for Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

For upgrades from Siebel 6.x to 7.7.x, decide whether or not to upgrade the revenues for nonprimary sales team members. Consider upgrading nonprimary sales team members' revenues if your organization does any of the following:

- Uses nonprimary forecasting and wants to continue using a similar nonprimary forecasting approach. (Running queries on the Revenue table may not be adequate to meet this need.)
- Wants each sales team member to have a different opinion on the close date and revenue amount for each opportunity.
- Wants to assign, to each sales team member, some portion of each opportunity that is not related to a specific product or some percentage of the overall opportunity value.

For each opportunity in previous versions of Siebel Business Applications, the upgrade evaluates the opportunity sales team records for nonprimary sales team members. Each sales team member record is used to create a revenue record, if the following is true:

- The primary flag is not checked.
- The revenue amount is nonzero.
- The commit flag is checked.

Before upgrading, make sure that your opportunity sales team records are appropriately marked as committed or not committed. Records marked as committed are automatically upgraded; records marked as not committed are not upgraded.

Typically, uncommitted sales team records are not upgraded.

## Preparing Siebel Products and Quotes Data for Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

### Customizable Products

Determine which of your Siebel 6.x configuration solutions are to be upgraded to Customizable Product quote items in Siebel 7.7.x. For Customizable Products, make sure that the Model Product field is checked in the Product Administration screen. This allows these products to be upgraded as Customizable Products.

For products that do not need to be upgraded as Customizable Products, verify that the Model Product field is unchecked. This verifies that quotes using these products in 6.x are upgraded as Packaged products.

All Siebel Configurator model associations for these products are removed, and these products behave like hierarchical product bundles. For more information on Packaged products and Configurator, see *Product Administration Guide*.

### Discount Amount Field

Read this section if you have implemented the Quotes line item Discount Amount Field so that it allows creation of a header-level discount when the field is either NULL or contains \$0.00.

The Siebel 6.x default behavior is to populate this field with \$0.00, and the user must then clear this before entering a header-level discount.

In Siebel 7.7.x, the default for this field is NULL. This means the user does not have to clear the field to create a header-level discount.

If you have altered the default behavior so that the user can create a header-level discount when the Discount Amount field is either NULL or contains \$0.00, you must run the following script on the Siebel Database before doing the upgrade:

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

# Preparing Siebel Financial Services Tables for Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development, production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

DESC\_TEXT is a column in two tables, S\_CONDITION and S\_PROD\_BNFT. To broaden platform support, this column is reduced in length from 250 to 200 characters during the upgrade from Siebel 6.x to Siebel 7.7.x.

Before the upgrade, examine these two tables for records that contain more than 200 characters. Manually reduce the size of these records to 200 characters or less.



# 12 Upgrading the Siebel Database

This chapter contains the following topics:

- [“Renaming the Siebel Repository” on page 203](#)
- [“Changing the Siebel Database Server Configuration Utilities Language” on page 204](#)
- [“Preparing to Run the Siebel Database Configuration Wizard” on page 205](#)
- [“Running the Siebel Database Configuration Wizard Under Windows” on page 208](#)
- [“Running the Siebel Database Configuration Wizard Under UNIX” on page 209](#)
- [“Starting the Siebel Upgrade Wizard” on page 211](#)
- [“Moving the Siebel Repository Files” on page 214](#)
- [“Regenerating SQL Files for a Siebel Upgrade” on page 215](#)
- [“Identifying and Dropping Obsolete Indexes for a Siebel Upgrade” on page 215](#)
- [“Preparing for a No-Development-Environment Siebel Upgrade” on page 217](#)
- [“Installing New Siebel License Keys During an Upgrade” on page 217](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Renaming the Siebel Repository

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

To prevent a naming conflict, before you run the upgrade, you must rename your existing development repository (*Siebel Repository*) to *Prior Customer Repository*. After the upgrade, your new development repository is given the name *Siebel Repository*.

When you rename the repository, you may prefix the name with additional characters (except for a leading space). Do not append the repository name with additional characters, because this results in an error.

### To rename the repository

- 1 Start Siebel Tools and connect to the Siebel database.  
Use the version of Siebel Tools for the Siebel release from which you are upgrading.
- 2 If you archived repository objects as .sif files, and you want to have them available in your application, import these archive files back into the repository.  
If you do not check these objects back into the repository, they will not be upgraded. You need only to check in those archived objects that you need in the future and want to have available in your upgraded application.
- 3 Choose View > Options.
- 4 Click the Object Explorer tab.  
The Object Explorer hierarchy displays.
- 5 Locate Repository in the list, put a check mark in the adjacent box, and then click OK.  
This exposes the repositories.
- 6 In the Object Explorer, click the Types tab, and then Click Repository.
- 7 In the Repositories list view, verify that your existing repositories do not use the names reserved for the upgrade process:
  - **New Customer Repository**
  - **New Siebel Repository**
- 8 Locate your current Siebel Repository in the list applet.
- 9 Click on the name and change it to Pri or Customer Repository.  
When you rename the repository, you may prefix the name with additional characters (except for a leading space, which causes the validation to fail).  
Do not append the repository name with additional characters, because this results in an error.  
For more information about renaming repositories, see *Configuring Siebel Business Applications*.  
**TIP:** Prefix the repository name with a date.
- 10 Step off the list to commit the record to the database.  
If the validation check fails, verify that the repository name is not prefixed with a leading space.

## Changing the Siebel Database Server Configuration Utilities Language

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Database Server Configuration Utilities launch in the language selected when you ran the Siebel Enterprise Server Installer. You can change the language in which the Utility runs, if desired, from the language chosen during installation.

To change the Database Server Configuration Utilities language, see the *Siebel Installation Guide* for the operating system you are using.

If you want an additional language to appear in the language list in the Database Server Configuration Utilities, you first need to install the appropriate language pack on the database server and on the Siebel Server. For information about installing additional language packs, see the *Siebel Installation Guide* for the operating system you are using.

### Related Topics

[“About the Siebel Database Server Configuration Utilities” on page 72.](#)

## Preparing to Run the Siebel Database Configuration Wizard

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Use this topic to identify the information you must enter when running the Database Server Configuration Utilities. Collect this information, and verify that it is correct before running the utilities.

The Database Server Configuration Utilities are a group of wizards that request information about the upgrade process you want to perform. They add this information to a master upgrade file and call an SQL generator. The SQL generator uses the information to create or populate SQL files:

- [Table 44 on page 205](#) lists the information that the utilities request for performing an upgrade.
- [Table 45 on page 208](#) lists the additional information specific to Prepare for Production mode.

After the Database Configuration Wizard exits you run the Upgrade Wizard. The Upgrade Wizard executes the SQL files against the Siebel database.

Table 44. Information Required for the Database Server Configuration Utilities

Field Name or Menu	Required Information
Gateway Name Server Address	The Siebel Gateway Name Server machine name.
Enterprise Server Name	The Enterprise Server name.

Table 44. Information Required for the Database Server Configuration Utilities

Field Name or Menu	Required Information
Siebel Server Directory	The absolute path of the directory where the Siebel Server is installed. For UNIX, do not enter the string \$SIEBEL_ROOT.
Siebel Database Server Directory	The absolute path of the directory where the Siebel Database Server is installed—for example: C: \sea7xx\dbsrvr.
RDBMS Platform	Choose the RDBMS type.
Database Version	Oracle only. Choose the database version.
Siebel Database Operation menu	For upgrep, upgphys and Prepare for Production modes, choose Upgrade Database.  The remaining menu choices are for database installation and administration.
Environment Type	Choose Development for development environment upgrades.  Choose Production for production test environment and production environment upgrades.
Upgrade Options	Choose one of the following: <ul style="list-style-type: none"> <li>■ Development Environment: <ul style="list-style-type: none"> <li>■ Upgrade Siebel Database Schema (upgrep)</li> <li>■ Upgrade Custom Database Schema (upgphys)</li> </ul> </li> <li>■ Production Environment: <ul style="list-style-type: none"> <li>■ Prepare for production</li> <li>■ Upgrade Siebel Database Schema (upgrep)</li> <li>■ Upgrade Custom Database Schema (upgphys)</li> <li>■</li> </ul> </li> </ul>
Siebel Industry Application	Choose the industry application you are upgrading from.  If you have upgraded to the base Siebel Business Application as part of upgrading to the new Siebel Industry Application release, choose Siebel Horizontal Application.
Current Siebel Version	Choose the application version you are upgrading from.
Database Encoding	Indicate whether your database uses a Unicode code page.
Language Selection	Upgrades with more than one language pack installed only. Choose the primary (base) language that is installed. The primary language is the language in which the data is stored in the Siebel database that is being upgraded.
ODBC Data Source Name	Verify the ODBC name for connecting to the Siebel database you are upgrading. If it is not correct, enter the correct ODBC name.

Table 44. Information Required for the Database Server Configuration Utilities

Field Name or Menu	Required Information
Database User Name and Database Password	Account name and password for the Siebel administrator of the Siebel database you are upgrading.
Database Table Owner and Database Table Owner Password	Account name and password for the Siebel database table owner.
Index Table Space Name and Table Space Name	Oracle and IBM DB2 only. Index table space name and table space name (4-KB table space name for DB2).
16-KB Table Space Name, 32K Table Space Name	IBM DB2 only. The 16-KB and 32-KB tablespace names.
Database Server OS	Choose the RDBMS server operating system type.
Parallel Indexing	<p>Oracle only. Select parallel indexing if you want SQL commands for index creation to include the arguments parallel, no logging.</p> <p>This causes an index to be created using parallel processing. This requires an RDBMS server with multiple CPUs. See your DBA to confirm if your RDBMS server is configured for parallel processing.</p> <p><b>TIP:</b> Oracle Library search phrase: parallel execution.</p> <p>Selecting parallel indexing does not cause multiple indexes to be created simultaneously, in parallel. To do this, you must set up parallel index-creation threads using Siebel Upgrade Tuner. You create parallel threads as part of tuning the production upgrade files. See <a href="#">Chapter 21, "Tuning the Siebel Upgrade Files."</a></p>
Log Output Directory	Accept the default or enter the directory name. If the directory does not exist, it will be created. Do not use special characters such as spaces or slashes.

### Additional Information Required for Prepare for Production Mode

Before performing the production upprep, you must run the Database Server Configuration Utilities in Prepare for Production Mode. The additional information shown in [Table 45](#) is required when you run the Database Server Configuration Utilities in Prepare for Production mode.

Note that several screens request information about the Siebel database in the development environment, not the production test or production environment.

Table 45. Additional Information Required for Prepare for Production Mode

Screen Name	Required Information
ODBC Data Source Name for Development Database	The ODBC name for connecting to the development environment Siebel database. If you are upgrading without a development environment, this is the ODBC name of the reference database.
Database User Name for Development Database	Account name and password of the Siebel administrator of the Siebel database in the development environment.
Database Table Owner for Development Database	Account name and password for the Siebel database table owner in the development environment.
Repository Name for Development Database	Enter the name of the upgraded Siebel Tools repository in the development environment database. Typically, this is Siebel Repository.

#### Related Topics

[“About the Siebel Database Server Configuration Utilities” on page 72.](#)

## Running the Siebel Database Configuration Wizard Under Windows

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Run the Database Configuration Wizard to upgrade the Siebel database. The Wizard collects information, populates a master configuration file, and calls the SQL generator to create SQL commands. The Upgrade Wizard then uses the configuration file and SQL commands to upgrade the Siebel database.

#### Prerequisites:

- Collect the information that the Database Configuration Wizard requires. See [“Preparing to Run the Siebel Database Configuration Wizard” on page 205.](#)
- Install the new release’s languages packs for all deployed languages. See [“Upgrade Planning for Multilingual Siebel Deployments” on page 129.](#)

### *To run the Database Configuration Wizard under Windows*

- 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 Select Start > Programs > Siebel SES Configuration > Configure DB Server.  
The first window of the Database Configuration Wizard appears.
- 3 Enter the information requested in each screen and click Next.
- 4 After you have entered all the requested information, the wizard displays a screen that lists the values you entered. If you need to make changes, click Previous.
- 5 When the window displays inquiring whether you want to start the Upgrade Wizard, do the following:
  - **Development Upgrep mode:** Answer No. Do not start the Upgrade Wizard. Instead, return to the upgrep process checklist and perform the next steps.
  - **Development Upgphys mode:** Answer Yes to start the Upgrade Wizard.
  - **Production Upgrep mode:** Answer No. Do not start the Upgrade Wizard. Instead, return to the upgrep process checklist and perform the next steps.
  - **Production Upgphys mode:** Answer Yes to start the Upgrade Wizard.
  - **Prepare for Production mode:** Answer Yes to start the Upgrade Wizard.

Just before displaying the prompt, the wizard calls the SQL generator to create or populate SQL scripts.

#### **Related Topics**

[“About the Siebel Database Server Configuration Utilities” on page 72.](#)

[“Starting the Siebel Upgrade Wizard” on page 211.](#)

## Running the Siebel Database Configuration Wizard Under UNIX

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** UNIX only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Run the Database Configuration Wizard to upgrade the Siebel database. The Wizard collects information, populates a master configuration file, and calls the SQL generator to create SQL commands. You then run the Upgrade Wizard to upgrade the Siebel database.

Prior to running the Database Configuration Wizard you must install the new release’s languages packs for each language that you deploy.

**Prerequisites:**

- Collect the information that the Database Configuration Wizard requires. See [“Preparing to Run the Siebel Database Configuration Wizard” on page 205](#).
- Install the new release’s languages packs for all deployed languages. See [“Upgrade Planning for Multilingual Siebel Deployments” on page 129](#).

**To run the Database Configuration Wizard under UNIX**

- 1 Verify that all servers are stopped:
  - Stop all Siebel Servers.
  - Stop the Siebel Gateway Name Server.
- 2 Make \$SIEBEL\_ROOT the current directory.
- 3 Source the environment variables from the siebsrvr root directory: *<install location>/siebsrvr*  
Korn shell (do not use a Bourne shell): `. si ebenv. sh`  
C shell: `source si ebenv. csh`
- 4 Review the values of the following environment variables and confirm the settings are correct:
  - SIEBEL\_ROOT. This path must end in siebsrvr. For example, `/usr/siebel/sea7xx/siebsrvr`.
  - LANGUAGE. This is the language in which the Database Configuration Wizard runs. The value of this variable is a language identifier string. For example, `enu` is the identifier string for English.

If either \$SIEBEL\_ROOT or \$LANGUAGE is not set or is incorrect, you must correct them before proceeding.
- 5 Start the Database onfiguration Wizard in either a C shell or Korn shell:  
`./bin/dbsrvr_config.ksh`
- 6 Enter the information requested in each screen and click ENTER.  
After you have entered all the requested information, the wizard displays a screen that lists the values you entered.
- 7 If you need to make changes, exit and rerun the script.
- 8 When prompted to start the Upgrade Wizard, do the following:
  - **Upgrep mode:** Answer No. Do not start the Upgrade Wizard. Instead, return to the upgrep process checklist and perform the next steps.
  - **Uppphys mode:** Answer Yes to start the Upgrade Wizard.
  - **Prepare for Production mode:** Answer Yes to start the Upgrade Wizard.

Just before displaying the prompt, the wizard calls the SQL generator to create or populate SQL scripts.

**Related Topics**

[“About the Siebel Database Server Configuration Utilities” on page 72.](#)

[“Starting the Siebel Upgrade Wizard” on page 211.](#)

## Starting the Siebel Upgrade Wizard

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Siebel Upgrade Wizard executes the upgrade of the Siebel database. It takes a master configuration file as input. This file contains environment information and a driver file name. The Upgrade Wizard executes the steps in the driver file to perform the upgrade.

As the Upgrade Wizard performs the steps in the driver file, it lists the steps in a state log. The state log is located in `/siebsrvr/LOG/process/state` where *process* is the upgrade process, for example `upgprep_prod_752` (upgrade from 7.5.2, upgprep process, production test or production environment).

If the Upgrade Wizard encounters an error and exits during an upgrade, you can restart it after correcting the error. The Upgrade Wizard reads the state log and continues the upgrade from the last successfully completed step.

When you run the Database Configuration Utilities, they will prompt you if you want to start the Upgrade Wizard.

### Prerequisites for Restarting the Upgrade Wizard

If the Upgrade Wizard stops due to errors, verify that you have met these prerequisites before restarting the wizard:

- Carefully review the relevant log files to make sure that your upgrade has completed successfully up to that point.
- 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 backup maintains a complete record of your log files, and prevents your previous log files from being overwritten, which could prevent accurate diagnosis of the reason for the break in the upgrade.

- If you are continuing a previous and incomplete schema upgrade, do not change the Log Output Directory that you previously selected.

- If problems with your environment prevent the upgrade from restarting, you must restore the database from the prior base version (the version from which you are upgrading). 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, delete or store the upgrade log files. The files are located in the following directory:

Windows: `SI EBEL_ROOT\log\PROCESS\output`

UNIX: `$SI EBEL_ROOT/log/PROCESS/output`

Also delete the state. log file. It is located in the following directory:

Windows: `SI EBEL_ROOT\log\PROCESS\state`

UNIX: `$SI EBEL_ROOT/log/PROCESS/state`

### Starting the Upgrade Wizard

Use this procedure to start the Upgrade Wizard. See the following topics to stop the Upgrade Wizard.

#### *To start the Upgrade Wizard*

- 1 Navigate to the following directory:

Windows: `SI EBEL_ROOT\bin`

UNIX: `$SI EBEL_ROOT/bin`

- 2 Enter the following command:

Windows: `si ebug /m master_UPGRADEOPTION_ENVIRONMENT_VERSION.ucf`

UNIX: `srvrupgwi z /m master_UPGRADEOPTION_ENVIRONMENT_VERSION.ucf`

where:

`_UPGRADEOPTION_ENVIRONMENT_VERSION` is the portion of the upgrade configuration file name that lists upgrade process, upgrade environment, and the Siebel Release from which you are upgrading. The file is located in `SI EBEL_ROOT\bin` (UNIX: `$SI EBEL_ROOT/bin`).

The following numbers are used for the Siebel Release portion of the file name:

- 621
- 703
- 704
- 752
- 77

■ 782

Table 46 lists an example of the file names for an upgrade from Siebel 7.5.x.

Table 46. Example of File Names for an Upgrade from Siebel 7.5.x

Upgrade Mode	File Name
Development env. upgrep	master_upgrep_dev_752.ucf
Development env. upgphys	master_upgphys_dev_752.ucf
Prepare for Production	master_prepare_for_production_upgrade.ucf
Production env. upgrep	master_upgrep_prod_752.ucf
Production env. upgphys	master_upgphys_prod_752.ucf

- 3 To begin the upgrade, click OK (Windows) or click ENTER (UNIX).

The Upgrade Wizard will notify you when the upgrade process is complete.

### Stopping the Upgrade Wizard Under Windows

Do not stop the Upgrade Wizard unless you are confident that an error has occurred, and the Upgrade Wizard or a utility it has called is hanging. Some SQL commands issued by the Upgrade Wizard or by its utilities can take considerable time to complete.

If you are not sure whether the Upgrade Wizard is hanging, contact Siebel Technical Support.

Stopping the Upgrade Wizard can have varying effects on the RDBMS. Before restarting the Upgrade Wizard, review the RDBMS log files. Run SQL commands as needed to resolve errors found in the RDBMS log files.

To stop the Upgrade Wizard, do one of the following:

- If the Upgrade Wizard has launched a separate command window in which a utility is running, close the command window. This terminates the utility and stops the upgrade.
- In the Upgrade Wizard dialog box, click Cancel.

The Upgrade Wizard will exit when the current upgrade step is complete. There may be a delay while the step completes in the RDBMS.

### Stopping the Upgrade Wizard Under UNIX

Do not stop the Upgrade Wizard unless you are confident that an error has occurred, and the Upgrade Wizard or a utility it has called is hanging. Some SQL commands issued by the Upgrade Wizard or by its utilities can take considerable time to complete.

If you are not sure whether the Upgrade Wizard is hanging, contact Siebel Technical Support.

Stopping the Upgrade Wizard can have varying effects on the RDBMS. Before restarting the Upgrade Wizard, review the RDBMS log files. Run SQL commands as needed to resolve errors found in the RDBMS log files.

### *To stop the Upgrade Wizard under UNIX*

- 1 If the Upgrade Wizard has started a utility in a child process, stop the child process.
- 2 Exit the shell in which the Upgrade Wizard is running.
- 3 Locate and stop any orphaned child processes started by the Upgrade Wizard.

After the processes terminate, there may be a delay while the RDBMS executes the issued SQL commands.

### Related Topics

[“About the Siebel Upgrade Wizard and Driver Files” on page 77](#)

## Moving the Siebel Repository Files

**Upgrades from:** All Siebel releases.

**Environments:** Development and production test.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Before doing the production test environment upgrade, you must copy the upgraded repository definition files (schema.ddl and custrep.dat) from the development environment to the production test environment.

**Prerequisite:** If you modified repository objects or schema definitions after completing the development upgrade, you must regenerate the schema.ddl and custrep.dat files. See [“Regenerating the Siebel Repository Definition Files” on page 240.](#)

### *To move the repository files*

- 1 In the development environment, navigate to the following directory:

Windows: *DBSRVR\_ROOT\PIatform*

UNIX: *DBSRVR\_ROOT/PIatform*

where:

*PIatform* is the database platform, for example, *DBSRVR\_ROOT\DB2UDB*.

- 2 Copy the following files:

custrep.dat

schema.ddl

- 3 In the production test environment, put these files in the following location:

Windows: *DBSRVR\_ROOT\PIatform*

UNIX: *DBSRVR\_ROOT/PIatform*

- 4 Make a copy of these files, and store them in a safe location.

**Platforms:** MS Windows, UNIX, IBM z/OS.

## Regenerating SQL Files for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development and production test.

If you enter incorrect information in the Database Server Configuration Utilities, the SQL files they generate will be incorrect and must not be used. You must run the Utilities again, enter the correct information, and regenerate the SQL files.

For example, you are upgrading a non-Unicode database. You run the Database Server Configuration Utilities and enter that you are upgrading a Unicode database. The SQL that the utility generates will be incorrect and cannot be used. You must regenerate the SQL files.

When the Database Server Configuration Utilities generate SQL files, they create a lock file that prevents regenerating the SQL files. To regenerate the SQL files, you must delete the lock file.

### *Regenerating SQL files with the Database Server Configuration Utilities*

- 1 Verify that the SQL files must be regenerated.  
Typically this is caused by entering incorrect information in the Database Server Configuration Utilities. If you are unsure if the files must be regenerated, contact Siebel Technical Support for assistance.
- 2 Navigate to `DBSRVR_ROOT\common` (UNIX: `DBSRVR_ROOT/common`) and locate the file `sql gen. usg`.  
This is the lock file that prevents the utility from regenerating the SQL files.
- 3 Delete the `sql gen. usg` file.
- 4 Rerun the Database Server Configuration Utilities.

### Related Topics

["About the Siebel Database Server Configuration Utilities" on page 72](#)

## Identifying and Dropping Obsolete Indexes for a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Use this topic to identify indexes that may be obsolete in the Siebel database and can be dropped. This topic is optional but is recommended since dropping obsolete indexes improves database performance.

When you run the Database Server Configuration Utilities in Prepare for Production mode, they do the following to identify obsolete indexes:

- Compares the repository schema definition in the development environment against the Siebel database physical schema definition in the production test environment.
- If an index is present in the Siebel database physical schema definition but not in the repository logical schema definition, the utility creates an SQL statement that drops the index. The utility places this SQL statement in a file.

This file is not executed when you run the Upgrade Wizard. You must manually review the file. If it contains indexes you want to drop, you must copy the corresponding SQL statements to another SQL file. This file is executed by the Upgrade Wizard after you run the Database Server Configuration Utilities in upgrep mode.

When the Upgrade Wizard runs, all indexes, including obsolete indexes, are maintained during table rebuilds and data migration. The obsolete indexes file is executed during the Create Siebel Indexes step.

**Prerequisites:** You must have run the Database Server Configuration Utilities in Prepare for Production mode in the production test environment.

### *To identify and drop obsolete indexes*

- 1 Navigate to the following file:

Windows: `DBSRVR_ROOT\platform\gen_obs_idx.sql`

UNIX: `DBSRVR_ROOT/platform/gen_obs_idx.sql`

where *platform* is the database type, for example DB2UDB.

- 2 Open the file with a text editor and review the SQL statements it contains.

The SQL statements drop indexes that are present in the Siebel database but not in the development environment repository logical schema definition.

- 3 If you want to drop an index, copy the corresponding SQL statement(s) to the following file:

Windows: `DBSRVR_ROOT\platform\obs_idx.sql`

UNIX: `DBSRVR_ROOT/platform/obs_idx.sql`

where *platform* is the database type, for example DB2UDB.

This file will be executed when you run the Upgrade Wizard after performing the database upgrep.

# Preparing for a No-Development-Environment Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If your installation does not include a development environment, you do not need to merge your Siebel Repository. Instead, you can use the repository and schema definition files included in the Siebel Database Server installation.

Before performing the upgrade, you must move and rename these files.

## *To prepare for a no-development-environment upgrade*

- 1 Navigate to *DBSRVR\_ROOT\common* (UNIX: *DBSRVR\_ROOT/common*) and locate the *mstrep.dat* file.
- 2 Copy the *mstrep.dat* file and rename it *custrep.dat*.
- 3 Place the *custrep.dat* file in the *DBSRVR\_ROOT\Platform* (UNIX: *DBSRVR\_ROOT/Platform*) directory, where *Platform* is the database platform, for example *DBSRVR\_ROOT\DB2UDB*.
- 4 In the *Platform* directory, copy the *ddl.ctl* file and paste the copy into the same directory.
- 5 Rename the copy *schema.ddl*.
- 6 In the production test environment create a new database, separate from the Siebel database. Install the Siebel database from the new release in the new database. Do not migrate any data to the new database. This database is called the reference database.
- 7 Define an ODBC for the reference database.

# Installing New Siebel License Keys During an Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

With the new release of Siebel Business Applications, you received one or more license keys. You must add all of the new license keys to enable the new release of the Siebel Business Applications, which you use in the next steps of the upgrade process.

For more information on installing license keys, see *Siebel Installation Guide* for the operating system you are using.

***To add new license keys***

- 1** Start the new-release version of Siebel Tools
- 2** Connect to the database server as the Siebel administrator.
- 3** Add your new license keys.

# 13 Reviewing the Siebel Upgrade Log Files

This chapter contains the following topics:

- “Summarizing Siebel Log Files Using the Logparse Utility” on page 219
- “Reviewing Siebel Upgrade Log Files for Errors” on page 221
- “Manually Archiving Siebel Upgrade Log Files” on page 226

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Summarizing Siebel Log Files Using the Logparse Utility

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** Windows and UNIX only. Does not apply to IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Use the Logparse utility to analyze and summarize the log files created when you run the Upgrade Wizard.

The Logparse utility writes its findings to a summary file:

Windows: `SI EBEL_ROOT\log\PROCESS\summary.html`

UNIX: `$SI EBEL_ROOT/og/PROCESS/summary.html`

where:

*Process* is the upgrade process you want to review, for example, `upgprep_dev_77`

The Logparse utility also writes a `summary.txt` file that contains the same information as `summary.html`. Use the `summary.txt` file if you do not have a browser.

After writing the output files, the Logparse utility opens `summary.html` automatically. If you do not specify an upgrade process, or if you specify `/r ALL`, you must open `summary.html` manually.

Table 47 lists the command line syntax for the Logparse utility.

Table 47. Command Line Syntax for the Logparse Utility

Flag	Parameter	Description	Comment
/s	install_dir	Full path to the parent directory of <i>SIEBEL_ROOT</i> (\$SIEBEL_ROOT). For example, if <i>SIEBEL_ROOT</i> is C:\sea7xx\si_ebsrvr, then <i>install_dir</i> is C:\sea7xx.  Enclose the path in quotes.	Required
/g	LANGUAGE_CODE	The language in which the Upgrade Wizard ran. This is called the resource language, and typically is the language in which you ran the Siebel Installation Wizard. For example, if the resource language is English, the language code is enu.	Required
/r	PROCESS	Name of the schema process for which you want to generate a summary file, for example upgrep_dev_77.  To obtain the schema process name, look in \si_ebsrvr\log. The subdirectory names in \log are the schema process names.	If you specify ALL, Logparse creates summary files for all the processes in \log.
/l	LOGPARSE_FILENAME	Name of the Logparse log file.	Default=logparse.log
/n	MAX_NUMBER_SQL	Maximum number of longest-running SQL to display in the summary.	Default value=10
/t	THRESHOLD_TIME	Threshold time for longest running SQL, in the format hh:mm:ss.	Default value=00:10:00  (By default, SQLs that run longer than 10 minutes are displayed)
/e	MAX_NUMBER_ERRORS	Maximum number of errors to display in the summary.	Default = 8

### To run the Logparse utility

- 1 Navigate to the following directory:

Windows: *SIEBEL\_ROOT*\bin

UNIX: \$SIEBEL\_ROOT/bin

- 2 Run the following command. Use the syntax in [Table 47](#):

```
logparse /s install_dir /g LANGUAGE_CODE /r PROCESS /l LOGPARSE_FILENAME
/n MAX_NUMBER_SQL /t THRESHOLD_TIME /e MAX_NUMBER_ERRORS
```

Windows example (Upgrade Wizard language is English):

```
logparse /s C:\sea7xx /g enu /r upgrep_dev_704 /l logparse.log /n 10 /t 00:00:10
/e 10
```

- 3 Review the Logparse log and verify that Logparse ran without errors.

Windows: `SI EBEL_ROOT\bi n\logparse.log`

UNIX: `$SI EBEL_ROOT/bi n/logparse.log`

### Related Topics

[“About the Siebel Database Upgrade Log Files” on page 80](#)

[“Reviewing Siebel Upgrade Log Files for Errors” on page 221](#)

## Reviewing Siebel Upgrade Log Files for Errors

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** Windows and UNIX only. Does not apply to IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Using the Logparse utility to review the log files created when you run Upgrade Wizard is the recommended way to verify that the upgrade process completed correctly and to identify errors that must be resolved.

The Logparse utility writes its findings to a summary file:

Windows: `SI EBEL_ROOT\log\process\summary.html`

UNIX: `$SI EBEL_ROOT/log/process/summary.html`

where

*process* is the upgrade process you want to review, for example `upgrep_dev_77`

Logparse also writes a `summary.txt` file that contains the same information as `summary.html`. Use the `summary.txt` file if you do not have a browser.

The procedures in this topic describe how to review the `summary.html` file.

## Verifying the Upgrade Environment

Before reviewing the upgrade steps for errors, verify that the upgrade environment is set up correctly.

### *To verify the upgrade environment*

- 1 If `summary.html` is not already open in your browser, navigate to `si_ebsrvr\log`, and locate the subdirectory for the upgrade process you want to review.
- 2 In the desired upgrade process directory, click `summary.html` to open the file in a browser.
- 3 On the `summary.html` home page, Upgrade/Install Log File Navigation Page, locate the Parameter column in the table, and click the Parameters link.

The Parameters page displays.

- 4 Review the parameters, and verify that they are correct.

License key values are encrypted. If you find errors in the parameters, correct them, and run the upgrade process again as needed.

## Determining If the Upgrade Process Completed Successfully

The upgrade processes consist of a series of steps. Each step must complete successfully.

If the Upgrade Wizard cannot complete a step, it marks the step incomplete in the state log (`si_ebsrvr\log\Process\state\state.log`) and exits. You must correct the error, and run the Upgrade Wizard again. When you rerun the Upgrade Wizard, it refers to the state log and resumes at the incomplete step that contained the error.

Use the following process to identify errors:

- 1 Resolve errors for steps with a Status of Incomplete.
- 2 Review all the steps with a Status of Complete. If any contain unacceptable errors, resolve these errors.
- 3 Restart the Upgrade Wizard, or if necessary, restore the database and rerun the upgrade process.

If you have any questions regarding how to resolve errors, contact Oracle's Siebel Technical Support.

### *To determine if the upgrade process completed successfully*

- 1 If `summary.html` is not already open in your browser, navigate to `si_ebsrvr\log`, and locate the subdirectory for the upgrade process you want to review.
- 2 In the desired upgrade process directory, click `summary.html` to open the file in a browser.
- 3 On the `summary.html` home page, Upgrade/Install Log File Navigation Page, locate the Step/Error column in the table and click the Steps/Errors link.

The Steps/Errors Summary page displays.

- 4 Locate the Status column, and verify that all the steps have a status of Complete.

If the Status of all the steps is Complete, the upgrade process completed successfully.

If the Status of any step is Incomplete, the upgrade process did not complete successfully. The Upgrade Wizard has exited before completing all the steps in the upgrade process. You must identify the error, and correct it.

**CAUTION:** In some cases, the Upgrade Wizard can complete a step even though the step contains unacceptable errors. You must verify that all steps do not contain unacceptable errors.

## Determining if Steps Contain Unacceptable Errors

There are two types of unacceptable errors:

- The Upgrade Wizard could not complete a step due to an error and exited. You can locate these errors by checking the Status field on the Steps/Errors Summary page. If the Status of a step is Incomplete, the step contains an unacceptable error.
- The Upgrade Wizard encountered an unacceptable error but was able to complete the step. You can locate these errors by examining the detailed information for each step.

### *To determine if steps contain unacceptable errors*

- 1 If `summary.html` is not already open in your browser, navigate to `si_ebsrvr\log`, and locate the subdirectory for the upgrade process you want to review.
- 2 In the desired upgrade process directory, click `summary.html` to open the file in a browser.
- 3 On the `summary.html` home page, Upgrade/Install Log File Navigation Page, locate the Step/Error column in the table, and click the Steps/Errors link

The Steps/Errors Summary page displays.

- 4 In the Name column, click the name link.

The detailed information for the step displays. An alternate method is to use the slide bar to browse down through the detailed information listing for all the steps.

- 5 In the detailed information for each step, check the Errors item.

If the value for Errors is None, the step has no errors.

- 6 If there is a table or a link to a table listing errors, open the errors file:

Windows: `DBSRVR_ROOT\DATABASE_PLATFORM\errors.rtf` or `errors.htm`

UNIX: `DBSRVR_ROOT/DATABASE_PLATFORM/errors.txt`

The errors file lists benign errors that do not need to be corrected.

- 7 Compare the errors listed in the Logparse summary.html file to the list of acceptable errors documented in the errors file.

If you find the error in the errors file, the error is acceptable and no action is required.

If you find an error that is not listed in the errors file, the error is unacceptable. You must correct the condition that caused the error

The log files generated by the Upgrade Wizard appear in the errors file as upgwi z1.log, upgwi z2.log, incrementing for additional log files.

- 8 To help resolve unacceptable errors, navigate to Siebel SupportWeb > Troubleshooting Steps > Product Areas > Upgrade, and choose Troubleshooting Steps 21. Common errors and how to resolve them are listed in this document. If the error is not listed or you cannot resolve it, contact Siebel Technical Support. Do not proceed with the upgrade.

## Determining if Other Log Files Contain Unacceptable Errors

Logparse summarizes the Upgrade Wizard logs. It does not analyze or summarize the logs of upgrade utilities called by Upgrade Wizard. For example, Logparse does not summarize the logs written by the ddlimp and dataimp utilities. You must review these logs for unacceptable errors.

### *To determine if other log files contain unacceptable errors*

- 1 Open the errors file:

Windows: `DBSRVR_ROOT\DATABASE_PLATFORM\errors.rtf` or `errors.htm`

UNIX: `DBSRVR_ROOT/DATABASE_PLATFORM/errors.txt`

The errors file lists benign errors that do not need to be corrected.

- 2 Navigate to the output directory.

Windows: `SIEBEL_ROOT\log\process\output`

UNIX: `$SIEBEL_ROOT/log/process/output`

where

*process* is the upgrade process you want to review, for example `upgprep_dev_77`

- 3 Using a text editor identify the logs that need to be reviewed.

The following logs can be ignored:

- ❑ Logs with names beginning `upgwi z` or `svrupgwi z`. They have either already been analyzed by Logparse, or they contain no useful data.
- ❑ Logs with names beginning `sw_cfg_XXX`.
- ❑ Logs with the name `siebel.log`.

- 4 Using a text editor open the logs and review them for errors.

- 5 Compare any errors to the list of acceptable errors documented in the errors file.  
If you find the error in the errors file, the error is acceptable and no action is required.  
If you find an error that is not listed in the errors file, the error is unacceptable. You must correct the condition that caused the error
- 6 To help resolve unacceptable errors, navigate to Oracle's Siebel SupportWeb > Troubleshooting Steps > Product Areas > Upgrade, and choose Troubleshooting Steps 21. Common errors and how to resolve them are listed in this document. If the error is not listed or you cannot resolve it, contact Oracle's Siebel Technical Support. Do not proceed with the upgrade.

## Reviewing Upgrade Performance

The Logparse summary file provides the input that Upgrade Tuner uses to analyze the performance of the upgrade scripts. For example, the summary file provides Upgrade Tuner with information about SQL commands that returned 0 rows.

If you plan to use the Upgrade Tuner to tune your production upgrade, familiarize yourself with the performance information in the summary file.

### *To review upgrade performance in the summary file*

- 1 If summary.html is not already open in your browser, navigate to `si_ebsrvr\log`, and locate the subdirectory for the upgrade process you want to review.
- 2 In the desired upgrade process directory, click `summary.html` to open the file in a browser.
- 3 On the `summary.html` home page, Upgrade/Install Log File Navigation Page, locate the following columns:
  - Performance Information for SQL
  - Performance Information for DDL
- 4 Click the links in these columns to display performance information about the upgrade process.  
The Performance Information for SQL page displays the longest running queries and queries that returned 0 rows for SQL scripts.  
The Performance Information for DDL page displays the tables created, altered, and deleted by the `ddlmp` utility and by SQL scripts.  
In both pages, the Net Cost column lists the time required to perform each operation.

### Related Topics

["About the Siebel Database Upgrade Log Files" on page 80](#)

["Summarizing Siebel Log Files Using the Logparse Utility" on page 219](#)

["Manually Archiving Siebel Upgrade Log Files" on page 226](#)

# Manually Archiving Siebel Upgrade Log Files

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** Windows and UNIX only. Does not apply to IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

After a successful installation and upgrade, you must manually save and archive the log files located in the *SIEBEL\_ROOT/Log/PROCESS* (Windows) directory.

By default, only nine upgrade 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 overwrites log files beginning with the earliest one created and recycles the rest as necessary. (This does not apply to the state.log file.)

The number of log files retained can be increased by resetting the `si_ebel_log_archive` environment variable to 20—for example, to retain twenty (20) log files.

# 14 Performing the Siebel Repository Merge

This chapter contains the following topics:

- [“Configuring Siebel Repository Objects to Inherit Upgrade Behavior” on page 227](#)
- [“Configuring Siebel Tools for the Repository Merge” on page 228](#)
- [“Performing a Siebel Repository Merge” on page 230](#)
- [“Reviewing the Siebel Repository Merge Log Files” on page 235](#)
- [“Generating Siebel EIM Temporary Columns” on page 237](#)
- [“Reviewing Siebel Repository Object Property Conflicts” on page 238](#)
- [“Regenerating the Siebel Repository Definition Files” on page 240](#)
- [“Setting Label Alignment for Siebel Text Fields” on page 242](#)
- [“Copying UI Files to a New Siebel Environment” on page 243](#)
- [“Deleting Unneeded Siebel Repository Files” on page 245](#)
- [“Running the Siebel Repository Preparation Wizard” on page 245](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Configuring Siebel Repository Objects to Inherit Upgrade Behavior

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, you can link objects together so that one object inherits the upgrade behavior of another. You do this by specifying an upgrade ancestor for an object.

You can specify an upgrade ancestor for the following object types:

- Applet
- Business component
- Integration object

■ Report

Before doing the repository merge, review new objects you have created and determine whether you want to specify an upgrade ancestor.

*To specify an upgrade ancestor*

- 1 Navigate to the object in Siebel Tools.
- 2 Click in the Upgrade Ancestor field.  
A dialog box appears. It lists available upgrade ancestors.
- 3 Select the desired upgrade ancestor and click Pick.

*To view the descendants or copies of an object*

- 1 Right-click on an object.
- 2 Select View descendants from the picklist.  
A dialog box appears and lists the descendants.

**Related Topics**

[“About the Siebel Repository Merge” on page 89](#)

[“About Inheriting Upgrade Behavior in a Siebel Upgrade” on page 94.](#)

## Configuring Siebel Tools for the Repository Merge

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

It applies to the Siebel Tools installation that you use to perform the repository merge.

If you are upgrading on Oracle 8i or 9i, you must run the development environment upgrade in Rule Based Optimization (RBO) mode. To support this mode, you must change the value of the SqlStyle parameter in the Siebel Tools configuration file.

If you are upgrading on Oracle 10g, you do not have to change the value of this parameter.

After the upgrade, if you are upgrading on Oracle 9i, you must reset the value to the default.

### Before the Repository Merge

Perform this task if you plan to run the development environment upgrade in RBO mode.

### ***To configure Siebel Tools before the repository merge***

- 1 Exit Siebel Tools.
- 2 Navigate to the Siebel Tools configuration file. It is located in the following directory:  
`install_dir\bin\lang\tools.cfg`  
where
  - `install_dir` is the Siebel Tools installation directory
  - `lang` is the language in which Siebel Tools is installed, for example enu.
- 3 Open the file with a text editor and locate the section that controls connectivity to the Oracle RDBMS.  
In most cases, this is the ServerDataSrc section.
- 4 In this section, set the SqlStyle parameter as follows: `SqlStyle = Oracle`.
- 5 Save the file and exit.
- 6 Restart Siebel Tools.

### **After the Repository Merge**

Do one of the following:

- If you are on Oracle 8i and want to continue using rule-based optimization (RBO), do not perform this task. The current setting of the parameter supports RBO.
- If you are on Oracle 8i and are using cost-based optimization (CBO), then perform this task. You must reset the parameter to support CBO.
- If you are on Oracle 9i, perform this task. You must reset the parameter to support CBO.

### ***To reset Siebel Tools configuration after the repository merge***

- 1 Exit Siebel Tools.
- 2 Navigate to the Siebel Tools configuration file you revised before the repository merge. It is located in the following directory:  
`install_dir\bin\lang\tools.cfg`  
where
  - `install_dir` is the Siebel Tools installation directory
  - `lang` is the language in which Siebel Tools is installed, for example enu.
- 3 Open the file with a text editor and locate the section containing the SqlStyle parameter that you revised.  
In most cases, this is the ServerDataSrc section.
- 4 In this section, set the SqlStyle parameter back to the default as follows: `SqlStyle = OracleCBO`.
- 5 Save the file and exit.

- 6 Restart Siebel Tools.

**Related Topics**

[“About the Siebel Repository Merge” on page 89](#)

# Performing a Siebel Repository Merge

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** Windows and UNIX only. Does not apply to IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This task merges customizations in the Prior Customer Repository (your current repository) into the New Customer Repository (the repository in the new Siebel release).

The repository merge typically takes five to seven hours to complete.

**Prerequisite:** The workstation on which the merge will occur must have a minimum of 512 MB of RAM and at least 2 GB of virtual memory or a 2-GB page file. Inadequate system resources is one of the most common causes of repository merge failures. If your repository includes a large amount of customization, additional memory may be required.

### *To merge the repository*

- 1 Verify that all Siebel Tools projects are checked in and unlocked.
- 2 Close network connections to all repository users and exit Siebel Tools.
- 3 Open the tool s.cfg file in the new Siebel release. It is located in the following directory:

*Tools\_install\_dir\bin\lang*

Where

*Tools\_install\_dir* is the directory in which Siebel Tools is installed, and *lang* is the language, for example enu.

- 4 Locate the SIEBEL section, and verify that the parameters are set as shown in [Table 48](#).

Table 48. Siebel Tools Configuration File Parameters for Upgrade

Parameter	Value
EnableToolsConstraint	FALSE
SymStrPrefix	X_

- 5 Save the file and close it.
- 6 Navigate to Control Panel > System > Advanced > Performance Settings > Visual Effects.

- 7 Click Adjust for best performance.
- 8 Start Siebel Tools in the new Siebel release, using the following command line option:  
*Tools\_install\_dir\bin\si ebdev\edi tseeddata.*

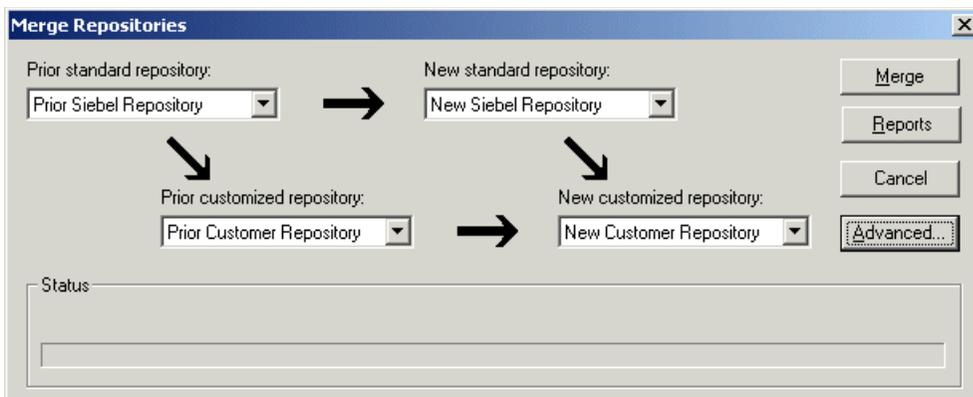
**NOTE:** The current install directory location for Siebel Tools should be *Tools\_install\_dir\bin*, as the *merge0.txt* file will be generated under that path.

- 9 Navigate to View > Options > Language Settings.
- 10 Verify that the language mode setting is set as desired.

This will be the user interface language for SRF files compiled from the New Customer Repository. It will also be the language used by the postmerge utilities.

- 11 Use the File > Open Repository command to open the Prior Customer Repository.
- 12 Choose Tools > Upgrade > Upgrade Application.

The Merge Repositories dialog box appears.



The Merge Repositories dialog box provides four options:

- **Merge.** This button merges the repositories you specify to produce a New Customer Repository.
- **Reports (for upgrades from Siebel 6.x only).** This button manually launches postmerge reports. For upgrades from Siebel 7.x, these reports are automatically generated after the repository merge.  
Do not select Reports unless you are upgrading from Siebel 6.x and you already completed your repository merge.
- **Cancel.** This button cancels the repository merge and exits the Merge Repositories dialog box.
- **Advanced.** This button opens the Merge Options dialog box described below.

13 In the Merge Repositories dialog box, choose the repositories listed in the following table.

Drop-Down List Item	Value to Choose
Prior Standard Repository	Prior x.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

14 Click Advanced.

The Merge Options dialog box appears.

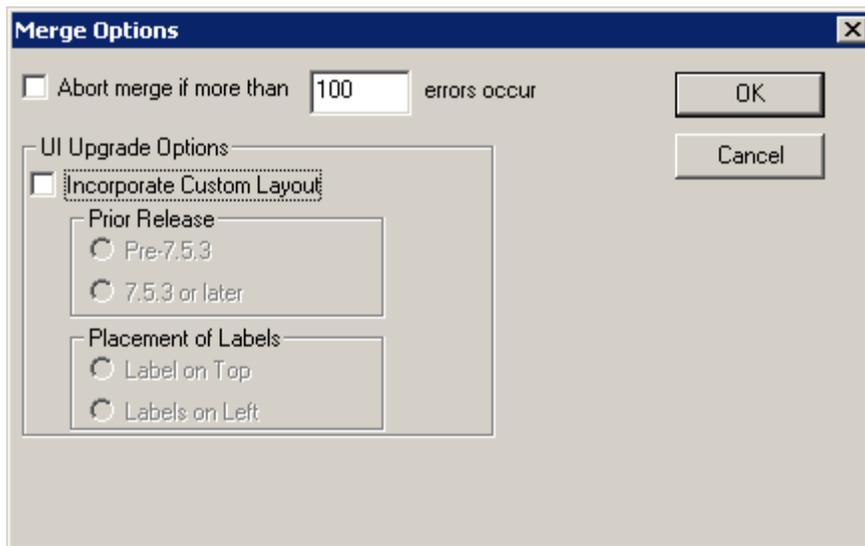
15 In the Merge Options dialog box, click the check boxes to activate or deactivate the merge options:

- **Abort merge if more than x errors occur.** Activate this option to abort the repository merge automatically if more than a designated number of errors occur.

**CAUTION:** The typical Repository merge generates many benign errors. If you select this option, set the number of errors to a large value. This will help prevent the Repository merge from aborting due to benign errors.

- **Incorporate Custom Layout.** Activate this option to help preserve field and button placement on prior custom or modified forms, views, and screens. Select a prior release and style for label placement.

The Prior Release is the release you are upgrading from. The placement of Labels controls where labels are placed in forms. As of Siebel 7.7, label alignment of fields changed. Label vertical alignment changed from top to middle, font weight changed from bold to normal, and text alignment changed from left to right. Select Labels on Top and Labels on Left to preserve the look and feel of releases prior to Siebel 7.7.

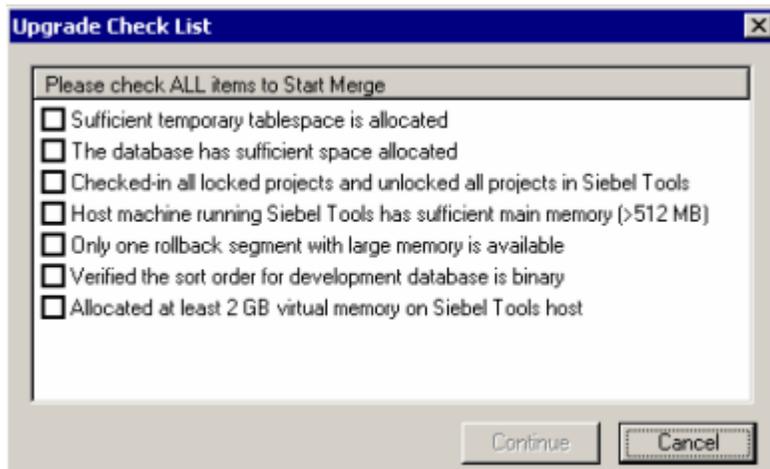


To continue, click OK.

The Upgrade Check List dialog box appears.

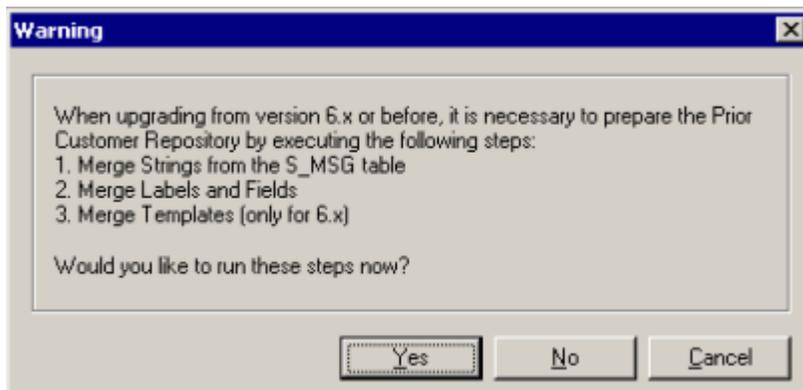
- 16 In the Upgrade Check List dialog box, verify that each prerequisite has been met. When they have been met, put a check mark in all the boxes and click Continue.

**CAUTION:** The upgrade may fail if all the items in the checklist are not completed.



A warning screen prompts you to prepare the Prior Customer Repository for the merge if you are upgrading from Siebel 6.x.

- 17 In the Warning screen, choose the option for your upgrade path.



- For upgrades from Siebel 6.x, click Yes to execute Prior Customer Repository preparation steps only if you have not already completed these steps. If you click Yes, the Prepare Repository Wizard takes you through these steps.
- For upgrades from Siebel 7.x, click No to skip the Prior Customer Repository preparation steps. These steps are not required.

- 18 Back up your development Siebel Database, which includes the merged repository, and click OK in the dialog box.

This launches the postmerge utilities. The postmerge utilities resolve common, merge-related user interface issues.

When the postmerge utilities start, the postmerge utilities dialog box displays the log entries from the utilities.

- 19 After the postmerge utilities complete, do the following:

- Click OK in the postmerge utilities dialog box.
- Close the merge options dialog box.

Do not click Merge in the Merge dialog box. This will restart the merge.

- 20 Back up the development environment database.

#### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Reviewing the Siebel Repository Merge Log Files

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

To determine if the repository merge was successful, review the merge log files. The merge is successful if it completes without unacceptable errors:

- **Acceptable errors.** If an ancestor object is specified in an object definition, and the ancestor object is not present in the New Siebel Repository, this causes a merge error. This is an acceptable error and can be ignored.

Here is an example of an acceptable error in the postmerge utilities log file, merge0.txt:

```
!!ERROR: : CANNOT upgrade objects which have Briefing Tracking Profile Appl et - Product marked as 'Upgrade Anc'
```

- **Unacceptable errors.** All other types of merge errors are unacceptable errors and mean that the merge was not successful.

For more detailed information, go to Siebel SupportWeb and search for “Troubleshooting Steps 19.”

Merge errors are displayed in the Upgrade Applications Objects List view in Siebel Tools. Additional details on merge errors are located in the repository merge log:

```
Tools_install_dir\bin\merge0.txt
```

where

*Tools\_install\_dir* is the directory in which Siebel Tools is installed.

Each time you run the merge process, the name of the merge0.txt file is incremented, for example merge1.txt.

If your repository merge process terminates and is flagged as “Incomplete” under Siebel Tools, Screens > Application Upgrader, the most common reasons for its failure are:

- The number of errors (!!ERROR) exceeds the number that was predefined in Siebel Tools when the merge was started.
- The merge process has been terminated due to a local issue on the Siebel Tools workstation, such as a scheduled reboot.
- RDBMS errors caused the process to stop.
- Memory allotment issues on the workstation on which Siebel Tools is installed.
- Network failure.

If the repository merge is terminated and flagged as 'Incomplete', this should be restarted as discussed in [“Performing a Siebel Repository Merge” on page 230](#).

### ***To determine if the repository merge was successful***

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 2 In the Application Upgrades list, select the record of the merge.
- 3 Check the entry in the Status column.
  - **Completed.** This means the merge completed without errors.
  - **Completed with Errors.** This means the merge contains errors.

If the Status column indicates Completed, no further action is required. The merge was successful.

If the Status column indicates Completed with Errors, you must review the errors to determine if the merge was successful. To review the errors, complete the remaining steps in this task.

- 4 In the Object Differences list, click Query.
- 5 In the Status field, enter ERROR::\* .
- 6 Press Enter to run the query.

This displays all the objects where the merge process encountered errors.

- 7 Open the merge log file, merge0.txt. It is located in the following directory:

`Tools_install_dir\bin`

where

`Tools_install_dir` is the directory in which Siebel Tools is installed.

If there are multiple files, open the one with the highest number in the file name, for example merge1.txt.

- 8 To locate merge errors in the file, search for the !!ERROR string.  
Additionally, informational messages are marked as !!INFO.
- 9 Use the objects displayed in the Object Differences list and the errors displayed in the log file to analyze the errors:
  - If all the errors are acceptable, the merge is considered successful.  
It is advisable, however, to consider the number of acceptable errors when determining whether to re-run the merge operation.
  - If the log contains unacceptable errors, the merge has failed.
- 10 If the merge contains unacceptable errors, go to Siebel SupportWeb and search for "Troubleshooting Steps 19."

This document explains the meaning of many of the error messages that can appear in the log file. Use this document to correct the errors. If you cannot resolve all the errors, use the directions in "Troubleshooting Steps 19" to contact Siebel Technical Support.

- 11 Open the workflow merge log file:

`Tools_install_dir\bin\merge0_ver.txt`

If there are multiple files, open the one with the highest number in the file name, for example merge1\_ver.txt. This log file is created by the workflow pre-merge and postmerge steps.

- 12 Review the log file. If the file contains errors, contact Siebel Technical Support for assistance.

### Related Topics

*Troubleshooting Steps 19: How To Troubleshoot Messages Generated During the Repository Merge in Siebel 7*, located on Siebel SupportWeb

["About the Siebel Repository Merge" on page 89](#)

## Generating Siebel EIM Temporary Columns

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The repository merge process does not preserve EIM processing columns for custom mappings. You must generate the missing custom EIM processing columns again.

### ***To generate EIM temporary columns***

- 1** In Siebel Tools, select File > Open Repository, and choose New Customer Repository.
- 2** Navigate to Tools > Upgrade > Generate EIM Processing Columns.  
A dialog box displays.
- 3** In the dialog box, click OK to generate EIM processing columns for custom mappings.
- 4** In the Object Explorer window, choose New Customer Repository and verify that the Comment field shows UpgEi mCol .

This indicates that the EIM temporary columns were created successfully.

### **Related Topics**

[“About the Siebel Repository Merge” on page 89](#)

## **Reviewing Siebel Repository Object Property Conflicts**

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

You can change how object property conflicts were resolved during the repository merge.

### **How Object Property Conflicts Occur**

The repository merge compares repository object properties in the Prior Siebel Repository, Prior Customer Repository, and New Siebel Repository. When the value of an object property is different in all three repositories, an object property conflict exists.

This occurs when you have changed the value of an object property in the Prior Customer Repository, and the value of this property has also changed in the new release (New Siebel Repository).

An object property conflict does not occur if you changed the value of an object property in the Prior Customer Repository, and the object property value did not change in the new release. When this happens, the merge process transfers the changed value to the New Customer Repository.

The merge process resolves object property conflicts by referring to the setting of the object’s Standard Win property. For about 90% of repository objects, the merge process resolves conflicts by using the object property value in the New Siebel Repository.

Do not change the setting of the Standard Win property.

## Application Upgrade Attribute List View

You can review and change how object property conflicts were resolved using the Application Upgrade Attribute List view in Siebel Tools. The Attribute Differences List in the view includes the following columns:

- **Object Name.** The name of the object.
- **Attribute.** The object property name.
- **Conflict.** The merge process puts a check mark in this field if there was an object property conflict during the merge.
- **Resolution.** Displays which property value the merge process used to resolve the conflict:
  - **Standard Value.** The property value in the New Siebel Repository was used. This value is displayed in the New Standard column.
  - **Custom Value.** The property value in the Prior Customer Repository was used. This value is displayed in the Prior Customized column.
- **Override.** Put a check mark in this column to change how the conflict is resolved. Overriding the resolution changes the property value in the merged repository. If the resolution was the Standard Value it is switched to the Custom Value and vice versa.

Putting a check mark in the Override column does not change the value displayed in the Resolution column. It indicates that the displayed value was manually overridden in the merged repository.
- **Prior Standard.** Displays the value of the object property in the Prior Siebel Repository.
- **Prior Customized.** Displays the value of the object property in the Prior Customer Repository. In the Resolution column, this value is called the Custom Value.
- **New Standard.** Displays the value of the object property in the New Siebel Repository. In the Resolution column, this value is called the Standard Value.

**Prerequisite:** The repository merge must have been successful. See [“Reviewing the Siebel Repository Merge Log Files” on page 235](#).

### *To review object property conflicts*

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Attribute List.
- 2 In the Application Upgrades list, select the record of the successful merge.
- 3 In the Attribute Differences list, click Query.
- 4 In the Attribute Differences list, click in the Conflict field so that a check mark appears.
- 5 Press Enter to run the query.

The query displays a list of all object properties for which there is a conflict.
- 6 For each record, review the entry in the Resolution field.

- 7 To change the resolution, click in the Override field.

A check mark appears. This changes the value of the object property in the merged repository.

Avoid overriding conflicts for the following object properties. Visually review these properties in the upgraded application before changing them:

- Left, right, top, height, width

#### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

## Regenerating the Siebel Repository Definition Files

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you have modified repository objects after the development environment upgrade (upgphys) and before upgrading the production test environment, you must regenerate the schema.ddl and custrep.dat files. These files were created during the upgphys:

- **Schema.ddl.** This file contains the logical definition of the Siebel database.
- **Custrep.dat.** This file contains the definition of repository objects.

These files are used as input to the production test and production environment upgrades. If you modify the object definitions or the schema definitions in the repository after these files have been created, you must regenerate the files.

### Regenerating the schema.ddl File

Use this procedure to regenerate the schema.ddl file.

#### *To regenerate the schema.ddl file*

- 1 On the Siebel Server where the Siebel Database Server files are installed, navigate to the following location:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 2 Run the following command:

```
ddl dict /u DatabaseOwner /p Password /c "ODBCDataSource" /d TableOwner /f  
DBSRVR_ROOT\DatabasePlatform\schema.ddl /e y /a y /l SiebelLogDir\sch_dict.log /n  
"Siebel Repository" /t dcir
```

Where

- *DatabaseOwner* is the Siebel database administrator account name.
  - *Password* is the Siebel database administrator account password.
  - *ODBCDataSource* is the ODBC name for connecting to the database. Enclose the name in quotes.
  - *TableOwner* is the Siebel table owner name.
  - *DBSRVR\_ROOT* is the absolute path to the Siebel Database Server installation directory.
  - *DatabasePlatform* is the Siebel Database Server directory name for the database, for example Oracle. The example shows Windows path syntax. On UNIX systems, use UNIX path syntax.
  - *SiebelLogdir* is the path to the directory where you want the output log placed (log output directory). The example shows Windows path syntax. On UNIX systems, use UNIX path syntax.
- 3 After the command completes, review the output log files for errors. If the log indicates there are errors, contact Siebel Technical Support.

## Regenerating the custrep.dat File

Use this procedure to regenerate the custrep.dat file.

### To regenerate the custrep.dat file

- 1 On the Siebel Server where the Siebel Database Server files are installed, navigate to the following location:

Windows: *SIEBEL\_ROOT*\bin

UNIX: \$*SIEBEL\_ROOT*/bin

- 2 Run the following command:

```
repimexp /a e /u DatabaseOwner /p Password /c "ODBCDataSource" /d TableOwner  
/r "Siebel Repository" /f DBSRVR_ROOT\DatabasePlatform\custrep.dat  
/l SiebelLogDir\exprep.log
```

where:

- *DatabaseOwner* is the Siebel database administrator account name.
- *Password* is the Siebel database administrator account password.
- *ODBCDataSource* is the ODBC name for connecting to the database. Enclose the name in quotes.
- *TableOwner* is the Siebel table owner name.

- DBSRVR\_ROOT is the absolute path to the Siebel Database Server installation directory. The example shows Windows path syntax. On UNIX systems, use UNIX path syntax.
  - *DatabasePlatform* is the Siebel Database Server directory name for the database, for example Oracle.
  - *SiebelLogdir* is the path to the directory where you want the output log placed (log output directory). The example shows Windows path syntax. On UNIX systems, use UNIX path syntax.
- 3 After the command completes, review the output log files for errors. If the log indicates there are errors, contact Siebel Technical Support.

### Related Topics

[“About the Siebel Database Upgrade Process” on page 68](#)

## Setting Label Alignment for Siebel Text Fields

**Upgrades from:** Siebel 7.x using ICL.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

At Siebel 7.7 label alignment for fields in forms changed. The label vertical alignment changed from top to middle, font weight changed from bold to normal, and text alignment changed from left to right.

If you selected Incorporate Custom Layout (ICL) for the repository merge and chose Label On Top, you must edit the style sheet (main.css) to set label alignment to preserve the look and feel of your previous release.

To ensure Quick Print and Print Preview display correctly, you must also revise the print preview style sheet (printmain.css).

### Revising main.css

Revise main.css to display field labels in pre-7.7 format.

#### *To revise main.css*

- 1 Using a text editor, open the main.css file in the Siebel Tools installation directory:

```
\publi c\<lang>\files\main.css
```

Where

*lang* is the installed language, for example enu.

- 2 Search for mceLabel.
- 3 In the .mceLabel section, set the following values as shown:
  - vertical-align : top
  - text-align: left
  - font-weight : bold
- 4 Search for mceLabel2.
- 5 Make the same changes in the .mceLabel2 section that you did in the .mceLabel section.
- 6 Save the file and exit.

### Revising printmain.css

Revise printmain.css to display field labels in pre-7.7 format in Quick Print and Print Preview.

- 1 Using a text editor, open the printmain.css file in the Developer Web Client:  
`\public\lang\files\printmain.css`  
Where  
*lang* is the installed language, for example enu.
- 2 Search for mceLabel.
- 3 In the .mceLabel section, set the following values as shown:
  - vertical-align : top
  - text-align: left
  - font-weight : bold
- 4 Search for mceLabel2.
- 5 Make the same changes in the .mceLabel2 section that you did in the .mceLabel section.
- 6 Save the file and exit.

#### Related Topics

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Copying UI Files to a New Siebel Environment

**Upgrades from:** Siebel 7.x using ICL. Also applies to non-ICL upgrades where you have manually revised Web template files or the style sheet file (main.css).

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

### ICL Upgrades

If you selected Incorporate Custom Layout (ICL) during the repository merge, a special set of Web template files was copied to the `webtempl` directory in the Siebel Tools installation used to perform the repository merge. For example, if you selected Siebel 7.5.3, new Web template files that provide the look and feel of 7.5.3 were copied to the `webtempl` directory.

You must copy these files to the Siebel Servers in the production test and production environments.

### Customized Web Template Files

When you customize Web template definitions using Siebel Tools, these changes are stored in the Siebel Repository. Siebel Tools does not make changes to the Web template files located in the Siebel Tools installation directory.

If you have manually customized the Web template files in the Siebel Tools installation directory in the development environment, you must copy these files to the Siebel Servers in both the production test and production environments.

### Copying Style Sheet Files to a Different Environment

When you customize properties of layout objects such as text fields using Siebel Tools, these changes are stored in the Siebel Repository. Siebel Tools does not make changes to the style sheet file (`main.css`) located in the Siebel Tools installation directory.

If you have manually customized `main.css` in the Siebel Tools installation in the development environment, you must copy this file to the Siebel Servers in both the production test and production environments.

You must also copy `main.css` to the new environment if you have done the following:

- Performed an ICL upgrade
- Selected Label on Top
- Edited the style sheet as described in [“Assigning an Item Identifier to Siebel Web Template Items” on page 279](#).

If you performed an ICL upgrade and selected Label on Top, you must also copy the style sheet for Quick Print and Print Preview (`printmain.css`) to the new environment.

#### ***To copy style sheet files to a different environment***

- 1 In the Siebel Tools installation directory (*installdir*) in the development environment, locate the style sheet files:

```
installdir \publ ic \lang\files\mai n. css and pri ntmai n. css
```

where *lang* is the installation language, for example enu.

- 2 Copy the style sheet files to the following location on the desired Siebel Servers in the production test or production environment:

Windows: `SIEBEL_ROOT\webmaster\files\lang`

UNIX: `$SIEBEL_ROOT/webmaster/files/lang`

To avoid overwriting the existing `main.css` and `printmain.css` file, consider changing the existing style sheet's filename before copying in the new one.

- 3 Update the Web server files.

This makes the new style sheet files available to the Web server. To update the Web server files, see *Siebel Installation Guide* for the operating system you are using. You can update the files using a URL and password or by stopping and restarting the Web server.

#### Related Topics

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

[“Upgrade Planning for Siebel Web Template Files and Style Sheet” on page 132](#)

## Deleting Unneeded Siebel Repository Files

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

When you are confident that the repository has been upgraded successfully, export the New Siebel Repository and Prior Customer Repository for safekeeping. You can also delete the following repositories:

- Prior Standard Repository
- New Standard Repository

#### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

## Running the Siebel Repository Preparation Wizard

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

When upgrading to Siebel 7.7.x, you must run the Repository Preparation Wizard on the Prior Customer Repository. You must do this before running the repository merge.

The Repository Preparation Wizard does the following:

- **Migrates strings.** Siebel 7.x supports locale strings in object-specific tables. The utility moves the S\_MSG data to the new locale table structure in Siebel 7.x.
- **Merges labels and fields.** Siebel 7.x merges labels with controls so that the label is a caption property of the control.
- **Merges Web templates.** Siebel 7.x merges Web templates in Siebel Tools to only use Base and Edit/Query/New for the majority of the preconfigured applets. Previous Siebel releases used four separate templates to deliver the same functionality.

### *To run the Repository Preparation Wizard*

- 1 In Siebel Tools, navigate to the Tools menu, and then choose Tools > Upgrade > Prepare Repository.
- 2 Choose Prior Customer Repository.  
The String Migration window appears.
- 3 In the String Migration window:
  - a Select the languages for which you want to migrate strings.
  - b Click in the check box beside Log migrated strings, then click Browse to specify the log file.  
If you have a single-language implementation and your base language is not English (ENU), you need to perform extra steps to upgrade labels of custom applets. See Oracle's Siebel SupportWeb Alert 426.  
To continue, click Next. The Merge Labels and Fields window appears.
- 4 In the Merge Labels and Fields window:
  - a Specify the input file by clicking Browse.  
The utility operates 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.  
If you are upgrading to Siebel Financial Services 7.x, do not use the default input file, applets.txt. Instead, browse to the Tools bin directory and select the input file named fi ns\_applets.txt.
  - b Click Next. The Merge Applet Web Templates window appears.
- 5 In the Merge Applet Web Templates window, click Browse to specify the same input file that you used in [Step 4](#), then click Next.  
The wizard prepares your Prior Customer Repository for the merge.
- 6 Verify that the wizard completed successfully:

- a In the Siebel Tools Object Explorer, click Repository.
- b In the Prior Customer Repository record, locate the Comments field.
- c Verify the field contains the following string: StMgLFmgTMrg.

If the string is present, the wizard completed successfully. If the string, or any part of it, is not present, the wizard did not complete successfully. Review the log files to locate the errors and contact Oracle's Siebel Technical Support.

7 Review the log files. They are located in the following directory:

*Tools\_install\_dir*\temp

Where

*Tools\_install\_dir* is the directory where Siebel Tools is installed.

The following table describes the log files.

File Name	Description
labelmerge.txt	Generated by label and field merge. Lists all applets that were modified. For each applet modified, displays whether the label and field merge was successful. If a control already had a caption, the old caption is listed for that control.
templatemerge.txt	Generated by the template merge. Lists all applets that were modified and displays whether the merge was successful. For each applet, displays the following: <ul style="list-style-type: none"> <li>■ The template used as the source (other templates are merged into this template).</li> <li>■ Applet Web Templates that were inactivated.</li> <li>■ Controls whose types were changed to ensure that they only appear in the correct modes.</li> </ul>

The following errors in the `templatemerge.txt` file are acceptable if they are against standard applets. This is because there were no Edit or New applet Web templates for standard applets in the previous Siebel release:

- Upgrading `APPLET_NAME`.
- Using Edit as source template.
- Could not find source template.



# 15 Migrating Siebel 6.x UI Customizations

This area contains the following topics. These topics apply only to upgrades from Siebel 6.x:

- [“About Migrating to the Siebel Web Client” on page 249](#)
- [“Identifying What Will Be Migrated in a Siebel 6.x Upgrade” on page 254](#)
- [“Exposing Hidden Properties of Siebel Applets and Views” on page 255](#)
- [“Running the Siebel Web Client Migration Wizard” on page 256](#)
- [“Reviewing Siebel Applet and View Migration” on page 259](#)
- [“Remigrating Siebel Applets and Views” on page 260](#)
- [“Regenerating the Siebel Postmerge Utilities Log” on page 261](#)

## Related Topics

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## About Migrating to the Siebel Web Client

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

As of Release 7.0.x, the Siebel Web Client is used as the deployment framework for all Siebel Business Applications. Siebel Web Client runs in a standard browser on the end user’s client computer. The browser connects through a Web server to the Siebel Server, which executes business logic and accesses data from the Siebel Database.

## Example of the Web Client UI

Figure 5 on page 250 shows a standard Siebel 6.x Windows Client. Figure 6 on page 251 shows a view that has been migrated from the Windows Client to the Web Client. This view is running in a browser.

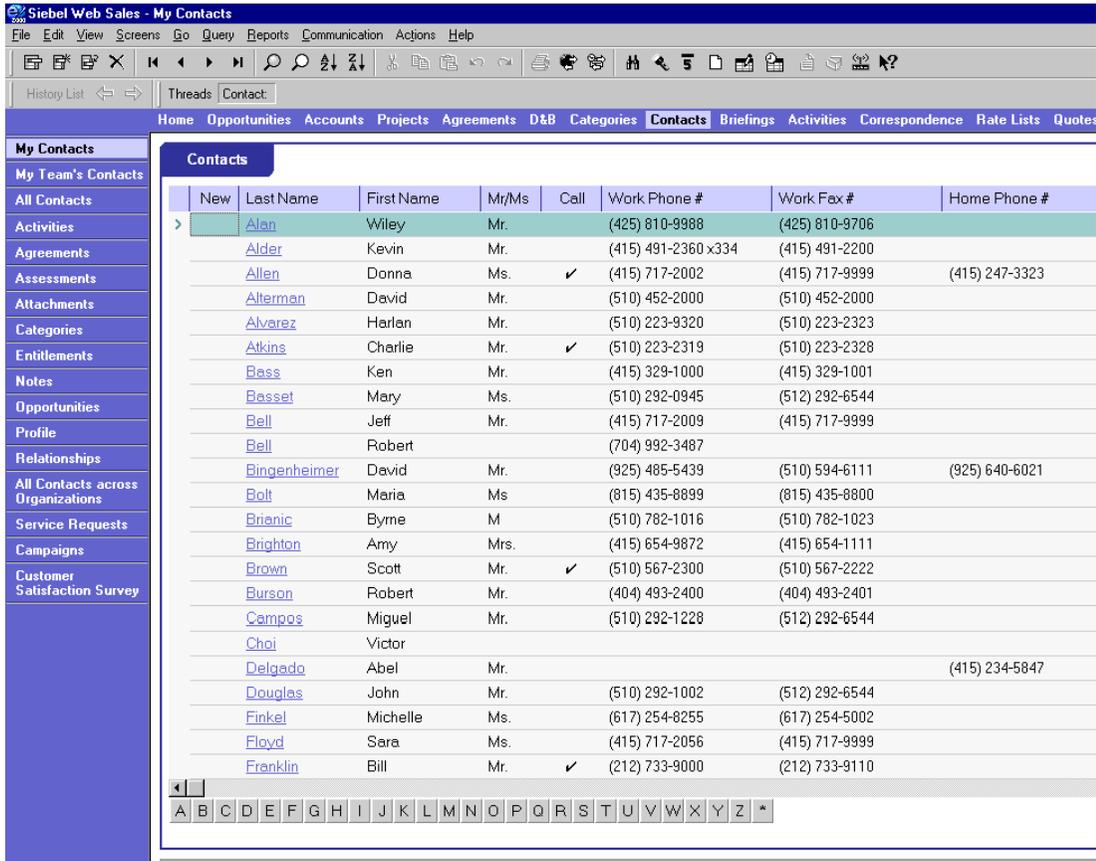


Figure 5. Contact List View—Windows Client

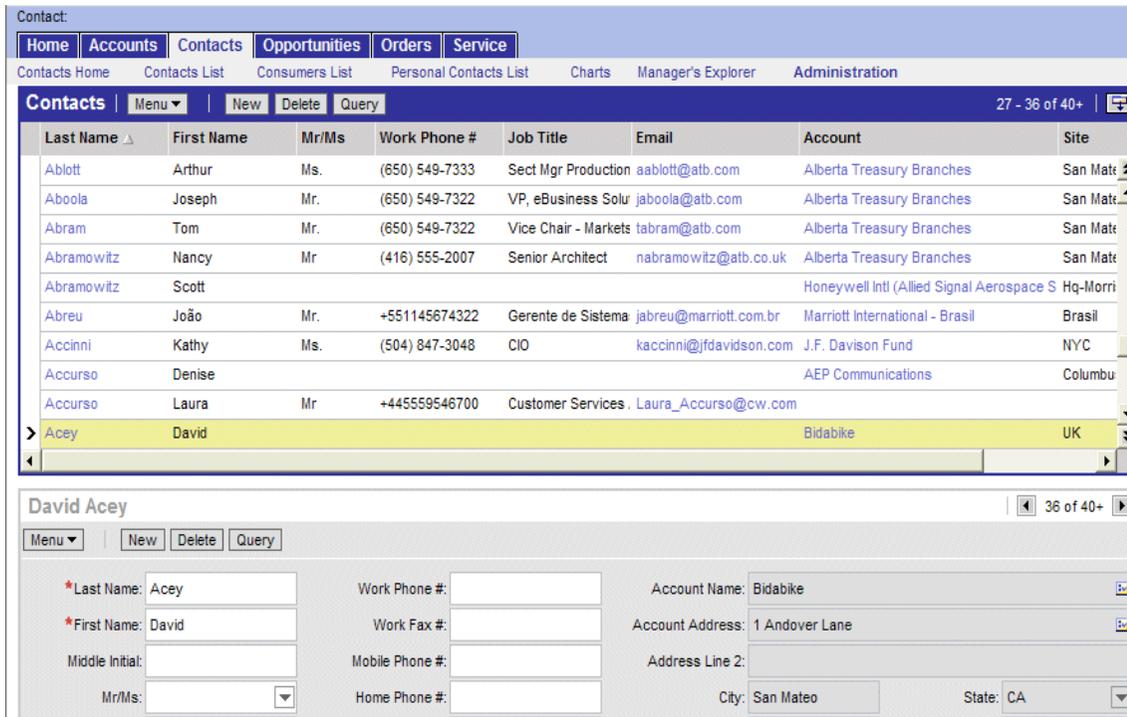


Figure 6. Contact List View—Web Client

## Web Client Migration Wizard

When you upgrade to Siebel 7.7.x, standard, uncustomized applets and views are automatically converted to Web-based layouts that can be rendered in a browser. Applets and views that you have created or modified are not converted automatically.

The upgrade does not convert applets or views that you have created or modified. To migrate created or modified applets and views, you must run the Web Client Migration Wizard. The wizard does the following:

- Identifies created or modified applets and views.
- Puts a comment in the Comments field of each. This identifies them so you can query for them.
- Provides a model applets and model views. These models provide Web templates that determine the layout of the applet or view.
- Migrates applets and views to a Web-based layout using the selected Web templates.

The Web Client Migration Wizard converts applets and views as a batch.

## Web Layout Wizard

The Web Layout Wizard allows you to migrate created or modified applets and views individually. You can select specific applets and views for migration. Then you select the desired model applet or model view to be used for the migration.

## Recommended Migration Strategy

The recommended strategy for using the two wizards is as follows:

- Run the Web Client Migration Wizard in Mark Applets and Views mode to identify which applets and views it will migrate. Review the marked applets and views, and select the desired model applets and model views.
- Expose hidden properties of applets and views. This improves the accuracy of the migration.
- Test the Web Client Migration Wizard against a few applets and views before running the wizard against the entire repository. Verify all model applets and model views work correctly.
- Migrate all the created and modified applets and views as a batch using the Web Client Migration Wizard.
- Review the migrated applets and views.
- Remigrate individual applets and views or groups of applets and views as needed using the Web Layout Wizard.

**NOTE:** Using the Web Layout Wizard as the only way migrate your applets and views is time-consuming and is not recommended.

## Model Applets and Views

The Siebel 7.7.x repository contains model applets and views. The Web Client Migration Wizard and Web Layout Wizard use these to migrate created and modified applets and views.

Before using model applets, compare the applets you intend to migrate with the layout of the model applets. Modify model applets as needed. For example, you can modify model applets to use different Web templates, controls, or control mappings.

Modifying model applets used for converting to grid-based layout is not recommended.

If you create new model applets, verify that you have defined all the Applet Web Template Items that you will need. Also, verify that you have assigned a unique HTML sequence number to all controls and list columns.

Before using model views, compare the views you intend to migrate with the layout of the model views. Modify the model views to meet your requirements.

Table 49 lists preconfigured model applets and views available in Siebel 7.7.x.

Table 49. Model Applets and Views

Model Applets and Views
Model chart applet
Model form applet
Model list applet
Model association list applet
Model MVG applet

Table 49. Model Applets and Views

Model Applets and Views
Model pick applet
Model tree applet
Model more applets view
Model one applet view
Model tree applet view

## How the Migration Is Done

The Web Client Migration Wizard and Web Layout Wizard map items from the dedicated client layout to the Siebel 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 *Configuring Siebel Business Applications*.
- **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 *Configuring Siebel Business Applications*.

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.
- **Buttons that appear on migrated applets.** 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.
- **Grid-based template and applet controls.** The grid layout conversion process converts a grid-based template and a set of applet controls with pixel-based coordinates into a Web-enabled layout that closely matches the original 6.x layout. For created or modified form applets that are converted to grid-based layout, you must manually map the buttons.

## Form Applets and Grid-Based Layout

Many form applets are converted to grid-based layout during upgrade. The following summarizes how grid-based applet Web templates differ from standard applet Web templates:

- You can modify the layout of the form using Siebel Tools without having to modify the Web template.
- With grid-based templates, labels and controls behave as separate items in the Web Layout Editor. This allows you to place them independently in the applet layout. However, labels and controls are really a single object in the repository with one set of shared properties.

- Grid-based templates do not automatically compress empty space in a column.

There are two applet Web templates that support grid layout. See [Table 50](#).

Table 50. Grid Layout Templates

Web Template	File Name	Comments
Applet Form Grid Layout	CCAppletFormGridLayout.swt	Use with all modes of form applets.
Applet Popup Form Grid Layout	CCAppletPopupFormGridLayout.swt	Use with all modes of popup form applets.

Not all form applets can be converted to a grid layout.

For a particular Applet Web Template mode, if the Web template type is *not* Applet Template—Form or Applet Template, the applet is not converted to grid layout.

If the Web template type is Applet Template—Form or Applet Template, but either the Applet name or the Web template name is specified in the awtcvctfg.txt file (located in the binary subdirectory of the root Siebel Tools installation directory), the applet is not converted to grid layout.

## Identifying What Will Be Migrated in a Siebel 6.x Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This topic describes how to use the Web Client Migration Wizard to identify the created and modified applets and views that will be migrated when upgrading to Siebel 7.7.x.

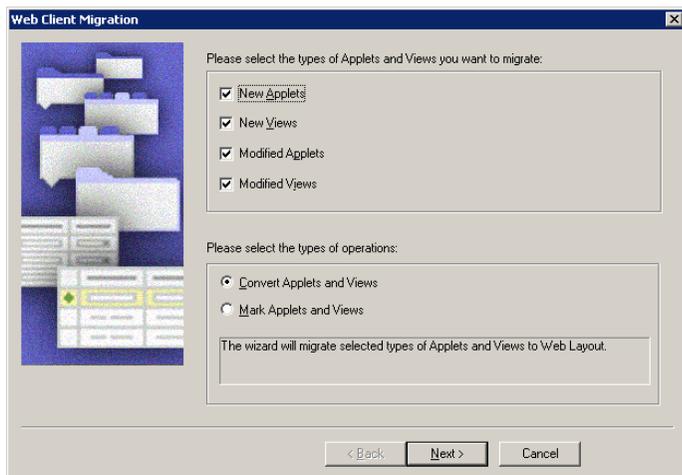
**Prerequisite:** The repository merge must have been completed successfully. See [“Reviewing the Siebel Repository Merge Log Files”](#) on page 235.

### *To identify which objects will be migrated*

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 2 In the Application Upgrades list, select the record of the merge.

- 3 Select Tools > Upgrade > Web Client Migration.

This starts the Web Client Migration Wizard. The model applet selection screen appears.



- 4 Select the types of applets and views that you want to migrate:

- New applets
- New views
- Modified applets
- Modified views

- 5 Select Mark Applets and Views and click Next.

No migration occurs.

Instead, the wizard puts TBMN (to be migrated new, for created applets and views) or TBMM (to be migrated modified, for modified applets and views) in the Comment field for applets and views to be migrated.

- 6 Query for objects with TBMN or TBMM in the Comments field to create a list of the objects that the Web Client Migration Wizard identified.

#### Related Topics

[“About Migrating to the Siebel Web Client” on page 249.](#)

## Exposing Hidden Properties of Siebel Applets and Views

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Release 7.0.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 7.7.x Web Client, expose these properties to make the migration of your customizations to Siebel 7.7.x easier.

### *To expose hidden properties*

- 1 Using a text editor, open the tools.cfg file located in the following directory:

*Siebel Tools* \bin\LANG

- 2 Under the [Siebel] section in the tools.cfg file, locate the following parameter:

ClientConfigurationMode

The default value for ClientConfigurationMode is Web. This value hides properties that are no longer applicable.

- 3 To expose these attributes for migration, change the value from Web to All.
- 4 Save and close the tools.cfg file.
- 5 Launch Siebel Tools.

The previously hidden properties are now visible.

- 6 After completing the Web Client migration to Siebel 7.x, reset the ClientConfigurationMode parameter to Web to hide properties that are no longer applicable.

### Related Topics

[“About Migrating to the Siebel Web Client” on page 249.](#)

# Running the Siebel Web Client Migration Wizard

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This topic describes how to use the Web Client Migration Wizard to batch-migrate your new and modified applets and views to Siebel 7.7.x.

### Prerequisites

- The repository merge must have been completed successfully. See [“Reviewing the Siebel Repository Merge Log Files” on page 235.](#)
- Identify model applets and model views. See [“Identifying What Will Be Migrated in a Siebel 6.x Upgrade” on page 254.](#) This prerequisite is optional.

- Expose hidden properties on objects. See [“Exposing Hidden Properties of Siebel Applets and Views” on page 255](#).

### Related Topics

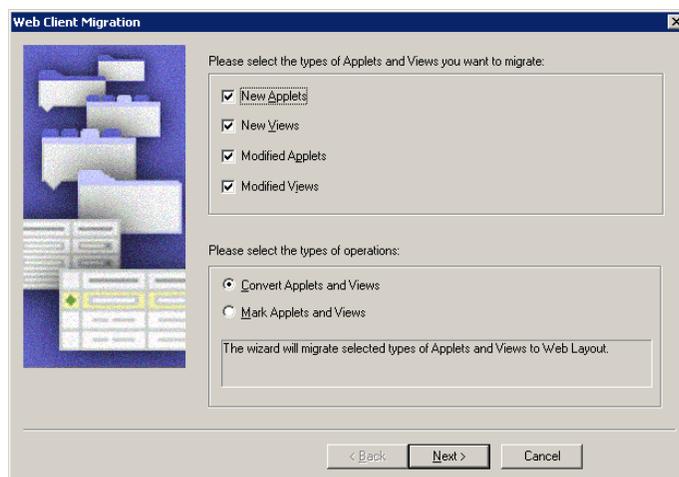
[“About Migrating to the Siebel Web Client” on page 249](#).

[“Remigrating Siebel Applets and Views” on page 260](#)

### To run the Web Client Migration Wizard

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 2 In the Application Upgrades list, select the record of the merge.
- 3 Select Tools > Upgrade > Web Client Migration.

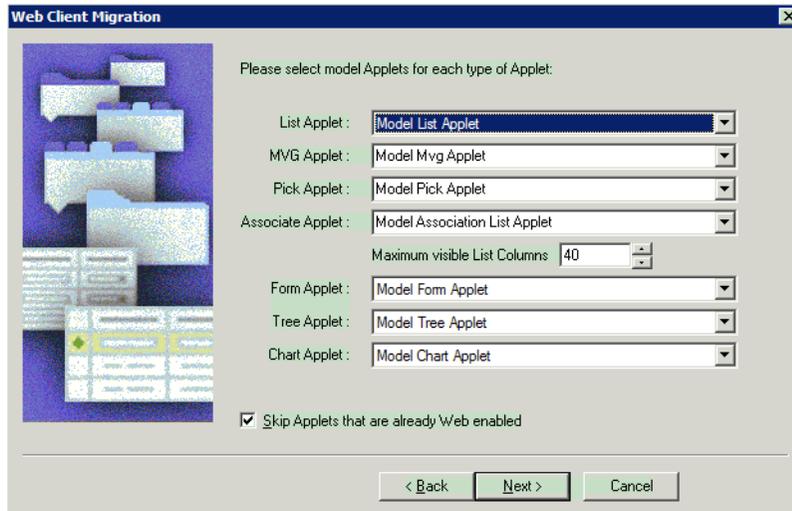
This starts the Web Client Migration Wizard. The model applet selection screen appears.



- 4 Select the types of applets and views that you want to migrate:
  - New applets
  - New views
  - Modified applets
  - Modified views
- 5 To continue, click Next.

6 Select a model applet for each applet type.

You can accept the recommended default values or select an alternate value.



By default, the maximum number of visible list columns is 40. Also by default, the option to Skip Applets that are already Web enabled is selected. This option preserves prior customizations if you have an existing customer application that uses Web templates.

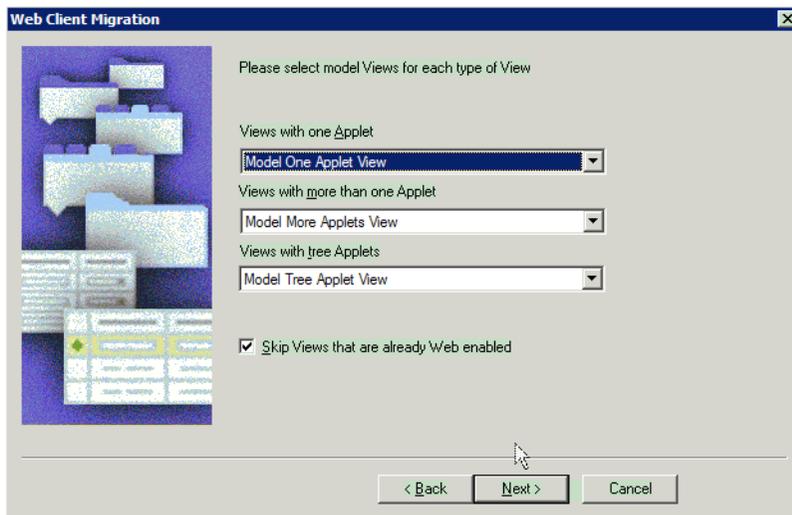
To continue, click Next.

7 Select a model view for each category of view:

- View with one applet
- View with two or more applets

■ Views containing tree applets

You can use any view of the appropriate category as a model view. The information used by the Web Client Migration Wizard includes the templates to be used and the applet modes to use for the View Web Template Items.



By default, the option to Skip Views that are already Web enabled is selected. This option preserves prior customizations if you have an existing customer application that uses Web templates.

8 To start the migration, click Next.

When the wizard completes, it appends the following to the Comments field of objects that it has migrated:

- **MigN**. Indicates new applets or views that have been migrated.
- **MigM**. Indicates modified applets or views that have been migrated.

## Reviewing Siebel Applet and View Migration

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Web Client Migration Wizard and Web Layout Wizard migrate new and modified applets and views to the Siebel Web Client. After running either wizard, review the migrated applets and views.

The following issues may require that you remigrate an applet:

- A set of nonfield controls is misplaced or absent in the Web layout.

- The template needs to be changed.
- Field controls are misplaced in the Web layout.

The following issues may require that you remigrate a view:

- The mode of a View Web Template Item is incorrect.
- The template needs to be changed.

Issues other than those above may need to be resolved manually.

### *To review applet and view migration*

- 1 In Siebel Tools, set the Object Explorer to Flat view and select Applet or View.
- 2 In the list view Comments field, query for Mi gN to review new applets or views. Query for Mi gM to review modified applets for views.
- 3 For applets, sort the resulting list of applets by Class.
- 4 For each applet or view, right-click on it and select Edit Web Layout.
- 5 Review the applet or view layout:
  - For applets, use the Applet Web Editor to make any necessary changes.
  - For views, use the View Web Editor to make any necessary changes.
- 6 To change the model applet or model view, run the Web Layout Wizard.

### Related Topics

[“About Migrating to the Siebel Web Client” on page 249.](#)

## Remigrating Siebel Applets and Views

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

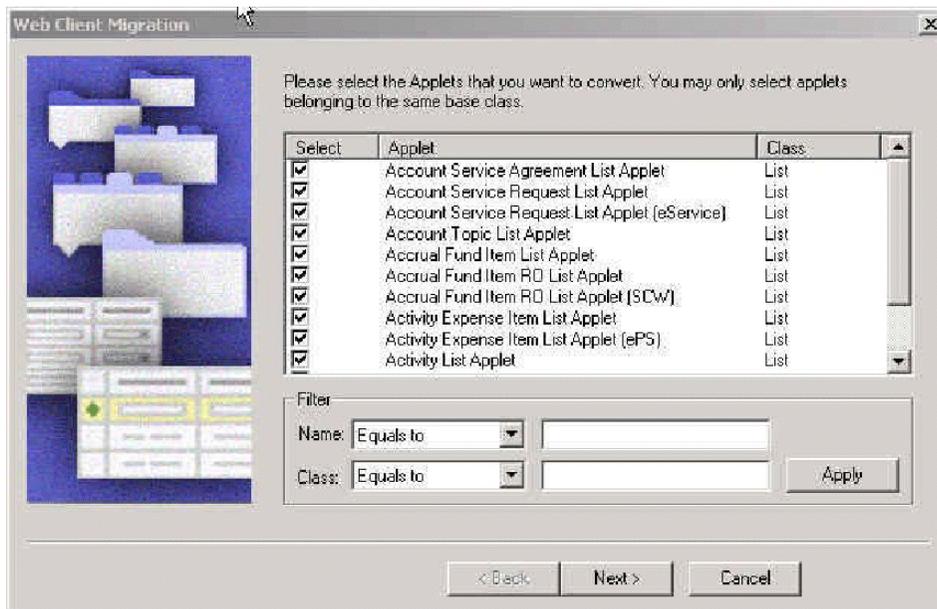
If some views or applets did not migrate properly when you ran the Web Client Migration Wizard to upgrade to Siebel 7.7.x, you can remigrate them using the Web Layout Wizard. With the Web Layout Wizard you can migrate applets or views individually or in groups.

When you select applets for migration, they must be in the same base class. When you select views, they must be of the same type.

You can run the Web Layout Wizard on applets and views multiple times.

**To run the Web Layout Wizard**

- 1 For applets, identify the desired model applet you want to use. For views, identify the desired model view.  
You may need to modify a model applet or model view to obtain the desired characteristics.
- 2 In Siebel Tools list view, select the applets or views you want to migrate.
- 3 In the list view, right-click and select Web Layout Wizard.



- 4 In the Web Client Migration dialog, verify that all the applets or views you want to migrate are selected. Click Next.
- 5 Select the model applet or model view you identified.
- 6 Click Next to migrate the applets or views.
- 7 Review the conversion results. Repeat the procedure if needed.

**Related Topics**

["About Migrating to the Siebel Web Client" on page 249.](#)

# Regenerating the Siebel Postmerge Utilities Log

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

When upgrading from Siebel 6.x to Siebel 7.7.x, you perform the repository merge and then the Web Client migration. This invalidates part of the postmerge utilities log created after the repository merge. You must regenerate the postmerge utilities log.

After the repository merge, the postmerge utilities start automatically. They help resolve common repository merge problems related to configuration and layout.

These utilities write to the postmerge utilities log:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\repreports.log
```

**Prerequisite:** You must have run the repository merge and the Web Client Migration Wizard.

### *To regenerate the postmerge utilities log*

- 1 In the Siebel Tools installation that you used to perform the repository merge, click Tools in the menu bar.
- 2 In the Upgrade submenu, click Upgrade Application.
- 3 In the Merge Repositories dialog box, set up the upgrade in exactly the same way as the actual upgrade, including the advanced settings.
- 4 Click Reports.
- 5 When the process completes, review the log.

# 16 Reviewing the Siebel User Interface

This chapter contains the following topics:

- [“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)
- [“Verifying Siebel Business Address Applet Configuration” on page 265](#)
- [“Reviewing Siebel Grid-Based Applets” on page 265](#)
- [“Reviewing Siebel UI Navigation” on page 267](#)
- [“Reviewing Siebel Multi-Value Group \(MVG\) Shuttle Applets” on page 268](#)
- [“Revising Siebel UI Rich Text Controls” on page 270](#)
- [“Reviewing New Siebel UI Aggregate Categories” on page 272](#)
- [“Revising Siebel Visibility Filters to Display Correctly” on page 273](#)
- [“Assigning a Category and Type to Siebel Chart Views” on page 274](#)
- [“Assigning a Category and Type to Siebel Explorer Views” on page 275](#)
- [“Setting Up Navigation to Inaccessible Siebel Detail Views” on page 276](#)
- [“Eliminating Obsolete Siebel UI Fields” on page 277](#)
- [“Reviewing Siebel UI Objects Affected by Incorporate Custom Layout” on page 278](#)
- [“Reviewing Required Fields in the Siebel UI” on page 279](#)
- [“Assigning an Item Identifier to Siebel Web Template Items” on page 279](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Troubleshooting Postmerge Siebel UI Problems

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

## Basic Troubleshooting Guidelines

If views or screens do not display, do the following:

- Using the Developer Web Client, verify that the user has been assigned the correct responsibilities.
- Query for the view or applet in the Administration-Personalization screen. Verify the conditional expression is correct. At Siebel 7.7 primary user role was replaced by primary user responsibility. See [“Upgrade Planning for Siebel Personalization” on page 150](#).
- In Siebel Tools, verify that the screen view and its parents have the `Display In Site Map` property, and the `Display In Page` property set to `TRUE`.
- Verify that the screen view and its parents have the `Viewbar Text` and `Menu Text` properties filled in.
- If an applet does not display all the fields or controls after upgrade, check in Siebel Tools for a Web template of the same name, but appended with `-Expanded` and specify this Web template for the applet. These templates provide additional placeholders for mapping fields and controls.
- In Siebel 7.5, the `Business Address` business component was replaced with the `CUT Address` business component. If you are upgrading from a release prior to 7.5, you must modify all links to `Business Address` applet so they point to `CUT Address`.

## About the Postmerge Utilities Log

After the repository merge, you must run the postmerge utilities. These utilities do the following:

- Validate UI components to verify they were migrated correctly to the new repository.
- Modify UI objects to implement new UI features. For example, they modify form applets and multi-value group applets to conform to the UI standard introduced at Siebel 7.7
- Verify that customized UI objects are configured correctly.

The postmerge utilities log lists the actions performed by the postmerge utilities. The log contains the following types of messages:

- **STATUS.** These messages provide information on the specific things that the postmerge utilities did. No action is required, so you can ignore these messages.
- **INFO.** These messages provide information on the specific things that the postmerge utilities did. No action is required, so you can ignore these messages.
- **WARNING.** These messages provide information on UI objects that may be incorrectly configured, so you should review them.
- **ERROR.** These messages indicate that a problem has been found that must be corrected.

The postmerge utilities log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### Related Topics

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

*Troubleshooting Steps 19: How to troubleshoot messages generated during the Repository Merge in Siebel 7*, located on Siebel SupportWeb

## Verifying Siebel Business Address Applet Configuration

**Upgrades from:** Siebel 6.x & 7.0.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic applies only to Siebel Industry Application (SIA) upgrades. It does not apply to Siebel Enterprise Application (HOR) upgrades.

At release 7.5, the Business Address business component was replaced with the CUT Address business component. You must modify all links to the Business Address applet so they point to CUT Address.

## Reviewing Siebel Grid-Based Applets

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Start Invalid Web Template Item Mapping Clean-up.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The postmerge utilities help convert many form applets to grid-based layout as of Siebel 7.7.

If a Prior Customer Repository applet that will be converted to grid-based layout has been customized by adding new fields or controls, the utilities put these fields and controls at the bottom of the applet in the New Customer Repository. After the repository merge, you must reposition these fields and controls.

**ICL upgrades only:** If Incorporate Custom Layouts (ICL) was selected for the repository merge, ICL converts many form applets to grid-based layout. The postmerge utilities then deactivate the grid-based layout and activate the Prior Customer Repository, flow-based form of the applet.

The reputility.log file lists applets that were converted by the postmerge utilities. Review the applets listed in the section of the log referenced above and revise layouts as needed.

The log is located here:

`SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputi l i t y . l o g`

Where

`SIEBEL_TOOLS_INSTALL_DIR` is the directory where Siebel Tools is installed.

Table 51 lists common issues and corrective actions for grid-based applets.

Table 51. Common Grid-Based Applet Issues

Log Examples	What To Do
STATUS:: Succeed Update AWTI List Mgmt Lists Entry Applet Base Status	The utility mapped a control to a different location. <b>Action:</b> Review the modified Applet. Verify that the new location for the control works.
STATUS:: Succeed Delete AWTI List Mgmt Lists Entry Applet Base Status ESN	To avoid possible overlapping, the utilities deleted a locale record for an applet Web template item. <b>Action:</b> Review the modified Applet Web Template and re-create the locale record.
WARNING:: Grid -> Grid merge/upgrade, Skip	Only flow-based form applets are converted to the grid-based Web template. <b>Action:</b> None required.
WARNING:: Upgrade Ancestor "Account Form Applet" Not Found in New Siebel Repository, Skip	The utilities did not find the upgrade ancestor's applet Web template in the New Siebel Repository. <b>Action:</b> Review the applet for invalid Web template items.
WARNING:: Applet Web Template Not Found in New Siebel Repository, Skip	The utility is trying to find the Applet Web Template from New Siebel Repository, but it is not found. <b>Action:</b> Review the applet for invalid Web template items.
WARNING:: No List Column Found, Applet "Program Expenditure List Applet", AWT "Edit", AWTI "Type", Control "Type", Skip	No List Column found for the Applet Web Template Item. <b>Action:</b> Delete the applet Web template item.
WARNING:: No Control Found, Applet "Expense Item Entry Applet", AWT "Edit", AWTI "Site", Control "Site", Skip	No Control is found for the Applet Web Template Item. <b>Action:</b> Delete the applet Web Template.
WARNING:: Button Control, Applet "Expense Item Entry Applet", AWT "Edit", AWTI "NewQuery", Control "NewQuery", Invoke Method "NewQuery", Skip	If a control has an Invoke Method action defined on it, the utility treats it as a button and will not remap it. <b>Action:</b> Review the applet and verify that the button control displays correctly.

**Related Topics**

- [“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)
- [“About Siebel User Interface Changes” on page 122](#)
- [“About the Siebel Postmerge Utilities” on page 97](#)
- [“About the Siebel Repository Merge” on page 89](#)
- [“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Reviewing Siebel UI Navigation

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “User Interface Navigation Upgrade.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The postmerge utilities analyze the repository and verify that objects referenced in screens, views and applets are defined correctly. The reputility.log lists objects that need to be modified. Review the section of the log referenced above and make the needed revisions.

The log is located here:

`SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log`

Where

`SIEBEL_TOOLS_INSTALL_DIR` is the directory where Siebel Tools is installed.

[Table 52](#) shows examples of common issues and corrective actions.

Table 52. Common UI Navigation Issues

Log Examples	What To Do
WARNING:: Project <i>projname</i> is not found in this Repository	The postmerge utilities exclude certain Tools projects from UI Navigation cleanup. One of the excluded projects was not found in the repository. <b>Action:</b> None required.
WARNING:: Ignoring Screen View Record. View Definition Not Found [View: Account Briefing View]	The utility is unable to read the view definition to determine where it should display. <b>Action:</b> Remove the invalid screen view record that references the invalid view.

Table 52. Common UI Navigation Issues

Log Examples	What To Do
WARNING:: Error Writing Category Record, Ignoring Changes [Name: <i>catname</i> ]	The utility could not update or insert a record in the Siebel database. Possible causes are that the record already exists or there is a database access problem.  <b>Action:</b> Verify that a duplicate <i>catname</i> record does not already exist, and then check database access.
WARNING:: Error Writing Screen View Record, Ignoring Changes [View: <i>viewname</i> ]	The utility could not create a screen view record. Possible cause is that the category does not exist. This error is often a consequence of an error updating or inserting a category record.  <b>Action:</b> Verify that the screen view category exists, and then check database access.

#### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

## Reviewing Siebel Multi-Value Group (MVG) Shuttle Applets

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Multivalue Group Shuttle Applet Upgrade.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, MVG applets are shuttle-enabled by default. The postmerge utilities shuttle-enable MVG applets in the New Customer Repository. This includes MVG applets from the Prior Customer Repository that you created or customized.

MVG applets must have a specific configuration in order to be enabled as MVG shuttle applets. For information on creating and managing MVG shuttle applets, see *Configuring Siebel Business Applications*.

In the reputility.log, the utilities list MVG applets that were converted. Review the section of the log referenced above and resolve any problems encountered during conversion.

The log is located here:

`SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log`

Where

`SIEBEL_TOOLS_INSTALL_DIR` is the directory where Siebel Tools is installed.

Table 53 lists common issues and corrective actions for MVG applets.

Table 53. Common MVG Applet Issues

Log Entry Example	What To Do
<p>WARNING:: [APPLET: Account Address Mvg Applet (NB)] MVG Applet is Inactive. Ignoring Applet.</p>	<p>This applet is inactive. <b>Action:</b> No action required.</p>
<p>WARNING:: [APPLET: Primary Employee Mvg Applet] [Applet Web Template: Base]  Applet Web Template is Inactive. Ignoring Applet Web Template.</p>	<p>The applet Web template is inactive. <b>Action:</b> No action required.</p>
<p>WARNING:: [APPLET: FINS Application Contact Mvg Applet - ACAPS]  [CONTROL METHOD INVOKED: ExecuteQuery] Has a Non Standard Control Type.</p>	<p>The utility is trying to map an existing Go (ExecuteQuery) button to the Edit List mode to enable Popup Inline Query. However, the Go button does not have the correct attributes. <b>Action:</b> Revise the control definition and map it to the Edit List mode.</p>
<p>WARNING:: [APPLET: State Model - State Mvg Applet]  [CONTROL METHOD INVOKED: UndoQuery] Has a Non Standard Control Type.</p>	<p>The utility is trying to map an existing Cancel (UndoQuery) button to the Edit List mode to enable Popup Inline Query. However, the Cancel button defined does not have the correct attributes. <b>Action:</b> Revise the control's definition and map it to the Edit List mode.</p>
<p>WARNING:: [APPLET: LOY Account Address Assoc Applet]  [CONTROL: CancelQuery] Control is Inactive. Please inspect and Reactivate.</p>	<p>The utility is trying to map an existing Cancel (UndoQuery) button to the Edit List mode to enable Popup Inline Query. However, the Cancel button defined is inactive. <b>Action:</b> Redefine a Cancel button and map it to the Edit List mode.</p>
<p>WARNING:: [APPLET: LOY Account Address Assoc Applet]  [CONTROL: ExecuteQuery] Control is Inactive. Please inspect and Reactivate.</p>	<p>The utility is trying to map an existing Go (ExecuteQuery) button to the Edit List mode to enable Popup Inline Query. However, the Go button defined is inactive. <b>Action:</b> Redefine a Go button and map it to the Edit List mode.</p>

Table 53. Common MVG Applet Issues

Log Entry Example	What To Do
<p>WARNING:: [APPLET: Account Address Mvg Applet] [Applet Web Template: Edit List]</p> <p>[Applet Web Template Item: ExecuteQuery] Applet Web Template Item is Inactive. Please inspect and Reactivate.</p>	<p>The utility is trying to map an existing Go/Cancel button to the Edit List mode to enable Popup Inline Query. However, the mapping already exists, but is marked inactive.</p> <p><b>Action:</b> Activate the mapping (Applet Web Template Item) in the Edit List mode and test.</p>
<p>WARNING:: [APPLET: Account Address Mvg Applet]</p> <p>[CONTROL: UndoQuery] Applet Web Template Item occupying Item Id 108. Cannot Map Control UndoQuery</p>	<p>The utilities tried to map a Cancel button to the default location 108 in Edit List mode. However, another control is already mapped to this location.</p> <p><b>Action:</b> Move the control at location 108 to another location, and then map the Go button to location 108.</p>
<p>WARNING:: [APPLET: Assoc Data Type Applet] Association List</p> <p>Applet contains both Base and Edit List. Manual review needed.</p>	<p>An MVG applet definition specifies both a base and edit-list Web template. At Siebel 7.7, the UI standard is that when MVGs first display, they are editable.</p> <p><b>Action:</b> For MVGs that users can edit, verify that the Web template specified for base mode and edit-list mode are the same. If not, change the base mode Web template so that it is the same as the edit-list template.</p>
<p>WARNING:: [APPLET: Activity Order Mvg Applet] [CONTROL: ExecuteQuery]</p> <p>Control is at an unexpected location. Expected Location is 107</p>	<p>The utilities tried to map a Go button to the default location 107 in Edit List mode. This is part of enabling Popup Inline Query. However, another control is already mapped to this location.</p> <p><b>Action:</b> Move the control at location 107 to another location, and then map the Go button to location 107.</p>

**Related Topics**

- [“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)
- [“About Siebel User Interface Changes” on page 122](#)
- [“About the Siebel Postmerge Utilities” on page 97](#)
- [“About the Siebel Repository Merge” on page 89](#)

## Revising Siebel UI Rich Text Controls

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 1: Rich Text Controls (RTC) That Need to Have Properties Reconfigured.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, the configuration of rich text controls (RTCs) changed. The user properties called RTC Graphic Field and RTC Link Field were moved from the Applet User Properties to the Control User Properties of the Body field.

The postmerge utilities review the repository and verify that rich text controls are defined correctly. The reputility log lists the controls that need to be modified. Review the section of the log referenced above and make the needed revisions.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### To revise RTC definitions

- 1 Start Siebel Tools. Set the Object Explorer to Flat.
- 2 Navigate to Views > Options > Object Explorer. Under applet, verify that Applet User Prop and Control User Prop are check-marked.
- 3 Refer to the reputility.log and query for one of the listed applets.
- 4 In the Object Explorer, select Applet User Prop.
 

If the user RTC Graphic Field and RTC Link Field user properties are marked Inactive, no further action is required.
- 5 For the following user properties, write down the value—active RTC Graphic Field and RTC Link Field user property:
  - RTC Graphic Field. The value for RTC Graphic Field typically is Body Field Graphic.
  - RTC Link Field. The value for RTC Link Field typically is Body Field Link.
  - RTC Body Field. The value is the control name.
- 6 For the applet, select Control > Control User Prop.
- 7 In the Controls list, query for the value you wrote down for RTC Body Field. This is the control name.
- 8 For the control, select Control User Prop.
- 9 Define the following control user properties on the control. Assign the values you wrote down for the applet user properties:
  - RTC Graphic Field
  - RTC Link Field

### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

## Reviewing New Siebel UI Aggregate Categories

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 2: New Aggregate Category Records That Should Be Renamed.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The following new properties were added to the Screen View object in Siebel Tools to support new UI navigation introduced in Siebel 7.7:

- Type (Aggregate Category, Aggregate View, Detail Category, Detail View)
- Parent Category
- Category Name
- Category Default View
- Display in Page
- Display in Site Map

For a description of these properties, see *Configuring Siebel Business Applications*.

If you have created new views or have modified existing views, the postmerge utilities create new Aggregate Category records to support the new properties. The utilities name the new aggregate category records “*busobj\_name* List.” For example, a new aggregate category record for the eEvents screen would be named eEvents List.

The reputility log lists the category records that were created by the postmerge utilities. Review the section of the log referenced above and make the following revisions as needed to the listed objects:

- Revise the Viewbar Text and Menu Text properties in all installed languages as required.
- Verify that the navigation hierarchy, including sequence numbers, is correct.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

## Revising Siebel Visibility Filters to Display Correctly

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 3: Views that need an applet in View Web Template Item ID 1.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel 7.7, the Show menu was replaced by a visibility filter. This drop-down menu lists alternative views for displaying records.

The postmerge utilities review the repository and verify that filters are defined correctly. The reputility log lists screens that have incorrectly defined filters. The most common problem is that none of the view Web template items has an Item Identifier of 1, which prevents the filter from displaying. Review the section of the log referenced above and make needed revisions.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### ***To revise visibility filters to display correctly***

- 1 Query for one of the views listed in the reputility.log.
- 2 For the view, select Base > View Web Template Item.
- 3 Set the Item Identifier for the first applet in the list to 1.
- 4 Refine the query to display the Parent Category listed in the log.

### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

# Assigning a Category and Type to Siebel Chart Views

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. Exception: this topic does not apply to 7.x upgrades that used the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** "Issue 4: Chart Views Needing Migration to Aggregate Type."

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

The Chart menu item is relocated as of Siebel 7.7 and displays as one of the aggregate categories under the view tabs. To ensure that the Chart menu item is located correctly, all the relevant charts for a view must be assigned to the same Aggregate Category. Also, each chart view must be of type Aggregate View. If the chart view is not of type Aggregate View, the chart menu item displays as a view tab.

The postmerge utilities review the repository and verify that Chart views have been defined correctly. The reputility log lists the screen views that require revision. Review the section of the log referenced above and make the needed revisions.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

## ***To assign a category and type to chart views***

- 1 Set Siebel Tools Object Explorer to Types.
- 2 Query for the screen.
- 3 In the Object Explorer, select Screen View.
- 4 Query for the screen views listed in the log.
- 5 Verify that all the views are assigned to the same category (Category Name).  
The Category Name can be null.
- 6 Verify that all the views are of type Aggregate View.

## **Related Topics**

["Troubleshooting Postmerge Siebel UI Problems" on page 263](#)

["About Siebel User Interface Changes" on page 122](#)

["About the Siebel Postmerge Utilities" on page 97](#)

["About the Siebel Repository Merge" on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Assigning a Category and Type to Siebel Explorer Views

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. Exception: this topic does not apply to 7.x upgrades that used the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 5: Explorer Views Needing Migration to Aggregate Type.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, explorer views are defined as Aggregate Views. In addition, explorer views use a new Web template, Tree 2.

The postmerge utilities review the repository and verify that explorer views are defined correctly. The reputility.log, lists the explorer views that require revision. Review the section of the log referenced above and make the needed revisions.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### ***To assign a new Web template to a view***

- 1 In Siebel Tools, query for a view name listed in the log.
- 2 In Object Explorer, select View Web Template.
- 3 In View Web Templates, change the Web Template to Tree 2.
- 4 Repeat for all the views listed in the log section.

### ***To assign a new type to a screen view***

- 1 In Siebel Tools, query for the screen.
- 2 In Object Explorer, select Screen View.
- 3 For the screen views listed in the log, change the Type to Aggregate View.
- 4 Repeat for each screen listed in the log section.

### **Related Topics**

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Setting Up Navigation to Inaccessible Siebel Detail Views

**Upgrades from:** All Siebel releases. Exception: this topic does not apply to 7.x upgrades that used the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 6: Categories Where Parent Applets Are Missing Drilldowns to a Detail View.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The postmerge utilities verify that you can navigate from a screen to all the screen detail views listed in the parent category for the screen. If any of the screen's detail views are not accessible using normal navigation methods, the utilities list the screen name, parent category, and the Aggregate View in the log.

In many cases, the problem is caused by a missing or incorrectly defined drilldown in a list applet in the view shown in the log. The missing drilldown prevents the user from navigating to a view containing third-level view tabs that provide access to all the detail views.

Review the section of the log referenced above and make needed revisions.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### ***To set up navigation to inaccessible detail views***

- 1 In Siebel Tools, navigate to the screen listed in the log. In Object Explorer, select Screen View.
- 2 In the Screen Views List, query for the following:
  - Type = Detail View
  - Parent Category = the Parent Category listed in the log
- 3 Start the application and navigate to the screen.

- 4 Try to navigate to the detail views listed in the query.

**TIP:** Use the Web Layout Editor in Tools to identify a detail view containing third-level view tabs that provide navigation to all the detail views. Verify that you can navigate to this view from a drilldown in the screen.

- 5 When you have identified the inaccessible detail view containing third-level view tabs, review the drilldown definitions in Siebel Tools for the list applet in the screen. Define a drilldown to the detail view if one does not exist.

#### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Eliminating Obsolete Siebel UI Fields

**Upgrades from:** All Siebel releases. This topic applies only to upgrades using the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 7: Fields deprecated from Business Components.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Some of the business component fields in your installed release may be obsolete in the new release. If you have customized existing applets or created new ones, and you selected ICL during the merge, the UI may contain obsolete fields or controls. If a business component field is not available for an applet field or control after the merge, the field or control does not display.

The reputility log lists the applets that contain obsolete business component fields. Review the section of the log referenced above and revise applet definitions and layouts as required.

The log is located here:

`SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log`

Where

`SIEBEL_TOOLS_INSTALL_DIR` is the directory where Siebel Tools is installed.

#### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

## Reviewing Siebel UI Objects Affected by Incorporate Custom Layout

**Upgrades from:** All Siebel releases. This topic applies only to upgrades using the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility log section:** “Issue 8: List of the items affected by PCL.”

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you selected Incorporate Custom Layout (ICL) for the merge, the postmerge utilities list all the UI objects that were affected by the ICL feature in the reputility log. Use this list to identify the screens, views, and applets you want to review when testing the UI.

The list has two parts:

- **“UI elements changed between prior Siebel Release and current Siebel Release.”** This part lists the screens, views, and applets that have UI changes in the new release. If you selected ICL during the merge, these changes are not reflected in the UI. Instead, the UI for the release you are upgrading from has been preserved.
- **“UI elements changed between prior Siebel Release and prior customer implementation.”** This part lists your customizations. The list includes screens, views, and applets you modified as well as those you created. These customizations are included in the merged repository. If you selected ICL for the merge, the UI for the customizations is the same as the release from which you are upgrading.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

### Related Topics

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

# Reviewing Required Fields in the Siebel UI

**Upgrades from:** All Siebel releases. This topic applies only to upgrades using the Incorporate Custom Layout (ICL) feature.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Reputility.log section:** "Issue 9: List of Required Fields Missing from the UI."

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

If you select Incorporate Custom Layout (ICL) during the repository merge, the Web templates for the UI you are upgrading from are used to display the UI after the merge. The Web templates for the new release are not used.

The Web templates used may not contain all the fields required by the new release. For example, a required field may be missing from the Web template used to display the Contacts applet. If you try to create a new record, the database will reject it because the record does not contain all the required fields.

This section of the reputility log lists all the applets that have missing required fields. Use the Web Layout Editor in Siebel Tools to add the required fields.

The log is located here:

```
SIEBEL_TOOLS_INSTALL_DIR\reppatch\log\reputility.log
```

Where

*SIEBEL\_TOOLS\_INSTALL\_DIR* is the directory where Siebel Tools is installed.

## Assigning an Item Identifier to Siebel Web Template Items

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

All View Web Template items must have a unique item identifier. Do the following to ensure this:

- See Technical Note 400 on Siebel SupportWeb and resolve duplicate item identifiers.
- In Siebel Tools, query View Web Templates for a null Item Identifier. Replace any null Item Identifiers with a number. If an item is part of a group of items that have the same grandparent (GParent), the number must be unique in the group.

### **Related Topics**

[“Troubleshooting Postmerge Siebel UI Problems” on page 263](#)

[“About Siebel User Interface Changes” on page 122](#)

[“About the Siebel Postmerge Utilities” on page 97](#)

[“About the Siebel Repository Merge” on page 89](#)

[“About the Siebel Incorporate Custom Layout \(ICL\) Upgrade Option” on page 100](#)

# 17 Postmerge Development Tasks for Siebel 6.x Upgrades

These topics apply only to upgrades from Siebel 6.x:

- [“About Migrating Siebel 6.x Scripts” on page 281](#)
- [“About Migrating Siebel 6.x Client-Side Interfaces” on page 283](#)
- [“Reviewing Siebel Customized Business Components” on page 283](#)
- [“Deleting Duplicate Siebel EIM Mappings” on page 284](#)
- [“Migrating Siebel Scripts Attached to Controls” on page 285](#)
- [“Migrating Siebel Scripts Attached to Applets” on page 287](#)
- [“Migrating Siebel Business Component, Business Service, and Application Scripts” on page 290](#)
- [“Migrating Outbound Siebel COM Interfaces” on page 291](#)
- [“Migrating Inbound Siebel COM Interfaces” on page 292](#)
- [“Setting Siebel Visibility Modes for Access Control” on page 295](#)

## Related Topics

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## About Migrating Siebel 6.x Scripts

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

You must migrate scripts on UI objects to server scripts or browser scripts when upgrading to Siebel 7.7.x.

### Server Scripts

Server scripts execute in the Application Object Manager. They existed in prior releases and continue to be supported in Siebel 7.7.x and later. These scripts are written in Siebel VB (for Windows platforms) and Siebel eScript (for Windows or UNIX). There are four types of server scripts:

- Business component server script
- Business service server script
- Application server script
- Applet server script

Server scripts enable you to script the behavior of business components, business services, applications, and applets by writing event handlers for events exposed by the scripting model.

## Browser Scripts

Browser scripts execute in 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. You can script the behavior of Siebel events as well as browser events that are exposed through the DOM.

Siebel Tools allows you to write scripts by selecting the appropriate User Agent. For information about browser script techniques, see Technical Note 386, available on Oracle's Siebel SupportWeb. For more information about events that are exposed through the DOM, see *Siebel Tools Online Help*.

You must migrate Siebel 6.x scripts written on applets to applet server scripts or business component server scripts, or you must rewrite them as browser scripts.

Applet script migration includes the following types of tasks:

- Moving Siebel 6.x applet scripts such as Control\_Click event handlers to business components
- Rescripting Siebel 6.x ActiveX controls in browser script
- Rewriting other Siebel 6.x applet scripts in browser script

To identify the number of 6.x applet scripts to be reviewed or migrated, look at the Object List Editor in Flat mode in Siebel Tools.

You must review Siebel 6.x scripts on the business components, business services, and application to identify references to methods, such as UI methods, that can not be accessed from server scripts.

To locate server scripts that you may need to modify, run the Siebel Tools Validator on business components, business services, and applications for which the Scripted flag is set to **True**.

You must modify Siebel 6.x business component and application scripts that contain references to UI methods. For example, you need to replace calls to MsgBox with calls to the RaiseError/RaiseErrorText method. Other modifications may include moving scripts to Browser script. For more information, see *Siebel Tools Online Help*.

## Error Handling

Two new methods are added in Siebel 7.x for obtaining application error information. These are errText and errCode. Revise the error handling portion of scripts to use these. For example,

```
{ ...  
    TheAppl i cati on(). Trace(e. toStri ng());  
    ... } ..."
```

should be changed to

```
{ ...  
    TheAppl i cati on(). Trace(e. errText);
```

```
TheAppl i cati on(). Trace(e. errCode);  
... } ..."
```

## About Migrating Siebel 6.x Client-Side Interfaces

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

Siebel Web clients support most of the events and methods on user interface objects supported by previous Siebel releases. In addition, the Web Client provides inbound and outbound client-side integration capabilities comparable to what was available in the business object interfaces in previous releases of the Siebel Dedicated Client. 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.
- **Inbound COM Interfaces.** In Siebel 6.x, desktop applications communicated with the Siebel Client through an Automation interface. In Siebel 7.7.x and later, the high interactivity Web Client provides similar automation interfaces.

This functionality is only available in a high interactivity framework, and it is applicable for the Siebel Web Client, Mobile Web Client, and Dedicated Web Client. (Unless otherwise specified, references in this section to the Siebel client may refer to any or all of these clients.)

## Reviewing Siebel Customized Business Components

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

After upgrading the Siebel Database to Siebel 7.7, thoroughly review the postupgrade configuration to make sure that the object level definitions are preserved as expected.

The access control buscomp migration utility, upgcust, is run on the Prior Customer Repository by the Upgrade Wizard during upgrade of the Siebel Database schema. This utility reconfigures custom business component configurations on obsolete tables. However, you may need to perform manual steps to business component fields and joins, depending upon the complexity of your business component configuration.

In some cases, not all custom buscomps are reconfigured by the access control buscomp migration utility. For example, customer buscomps that are based on customer extension columns or on Siebel columns in obsolete tables need to be identified and manually fixed after the upgrade.

During the upgrade, the access control buscomp migration utility, upgcust, generates a log file titled upgcust.log that contains a list of business component joins and fields that need to be manually rectified. This particular log file, along with others generated by the upgrade process, can be found under *SIEBEL\_ROOT\log* (Windows).

## Log File 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

## Log File Part 2

After the repository merge has been run, there may be inconsistencies in the join names and joins set at the field level due to merge behavior. If you do not resolve these discrepancies, the application configuration may result in errors or 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 business component name and the join name.

With this list, you must 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, perform any manual configuration in the New Customer Repository after the upgrade has been run. For more information about working with Business Component definitions and joins, see *Siebel Tools Online Help*.

# Deleting Duplicate Siebel EIM Mappings

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Any custom Enterprise Integration Manager (EIM) mappings that were added to previous versions of Siebel Business Applications are automatically migrated to the new custom repository. If custom mappings are duplicated by new mappings added to Siebel 7.x, errors occur when you use EIM.

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.

Upgrading the database schema generates a file called `mapclash.txt`. This file lists any custom EIM mappings that conflict with Oracle-provided mappings. The file contains the columns illustrated and defined in [Table 54](#).

Table 54. Mapclash.txt File Column Names and Definitions

Column Name	Definition
IT_Name	Interface table name (example: S_ACCOUNT_IF)
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.

## Migrating Siebel Scripts Attached to Controls

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Release 7.0.x, the `ControlName_Click` event handler was replaced by the `Method Invoked` property on control objects. By specifying the `Method Invoked` property, when a user clicks on a control (for example, a Button), the client framework checks to see if the method was implemented in browser script associated with the `Applet_PreInvokeMethod` event.

If the method was not implemented in browser script, the request is sent to the server for execution. After this request is on the server, the Object Manager verifies that the method was implemented in the WebApplet\_PreInvokeMethod event, and then the BusinessService\_PreInvokeMethod event.

If the method was not implemented in either browser or server script, an error occurs.

Control methods that were formerly available in Siebel VB or Siebel eScript also are no longer available. In Siebel 7.x, all control methods are available through browser scripting and execute in the high interactivity applications only. [Table 55](#) provides a list of the control methods that are supported by browser script. For more information, see *Siebel Tools Online Help*.

Table 55. Control Method Syntax Summary

Method	Description	Syntax
Applet	Returns the parent applet for the control.	var oControl; var oApplet; oApplet = oControl.Applet();
Buscomp	Returns the corresponding business component for the control.	var oControl; var busComp; busComp = oControl.Buscomp();
GetValue	Returns the value of a control.	var oControl; var sValue; sValue = oControl.GetValue();
Name	Returns the name of the control.	var oControl; var sName; sName = oControl.Name();
SetValue	Sets the contents of the control to the indicated value.	var oControl; oControl.SetValue( <i>value</i> );

In addition to specifying the Method Invoked property, controls may also be scripted based upon native DOM events supported by the browser.

Figure 7 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.

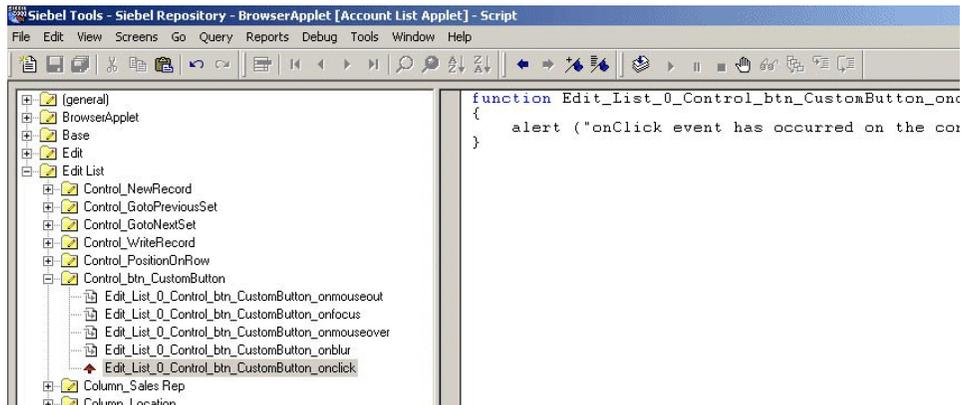


Figure 7. DOM Events When MiniButton Control Added to Applet

### Related Topics

["About Migrating Siebel 6.x Scripts" on page 281.](#)

["Migrating Siebel Scripts Attached to Applets" on page 287](#)

["Migrating Siebel Business Component, Business Service, and Application Scripts" on page 290](#)

## Migrating Siebel Scripts Attached to Applets

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

When upgrading to Siebel 7.7.x, you must do one of the following:

- Migrate Siebel 6.x scripts written on applets to applet server scripts or business component server scripts. Migrating applet scripts includes the following types of tasks:
  - Moving Siebel 6.x applet scripts such as Control\_Click event handlers to business components
  - Rescripting Siebel 6.x ActiveX controls in browser script
  - Rewriting other Siebel 6.x applet scripts in browser script
- Rewrite Siebel 6.x scripts as browser scripts.

To identify the number of 6.x applet scripts to be reviewed or migrated, look at the Object List Editor in Flat mode in Siebel Tools.

As shown in [Figure 8 on page 288](#), applet scripts implemented in Siebel 6.x 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

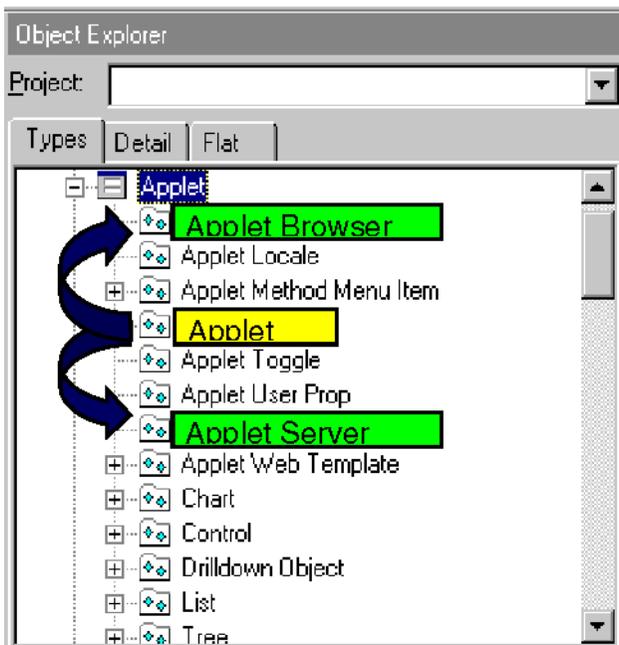


Figure 8. Movement of Applet Scripts

In Siebel 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 is straightforward, because several WebApplet events correspond to Siebel 6.x applet events. Corresponding browser or server script event handlers might be available, but the scripts are not automatically migrated. Table 56 depicts Siebel 6.x events and corresponding Siebel 7.x events available in either browser or server script.

Table 56. Siebel 6.x Applet Events and Siebel 7.x Equivalents

Siebel 6.x Applet Event	Siebel 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 Siebel 7.7.x events (for example Applet\_LostFocus event), you must evaluate and reimplement the functionality of the Siebel 6.x script. 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 Siebel 7.x. The obsolete applet methods are listed in Table 57. In some cases, the methods were reimplemented as browser script methods, as in the case of FindControl and FindActiveX Control. Other methods, such as PostChanges, were replaced by properties that can be configured in Siebel Tools (Immediate Post Changes property on a Field). A few other methods require modifications to scripts that use these methods.

Table 57. Obsolete Applet Methods

Applet Method
ActiveControl
FindControl
FindActiveXControl
GotoControl
PostChanges
Drilldown

### Related Topics

[“About Migrating Siebel 6.x Scripts” on page 281.](#)

[“Migrating Siebel Scripts Attached to Controls” on page 285](#)

[“Migrating Siebel Business Component, Business Service, and Application Scripts” on page 290](#)

## Migrating Siebel Business Component, Business Service, and Application Scripts

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

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. If existing scripts refer to methods that became obsolete or contain references to the Siebel user interface, the scripts need to be edited.

The following application and business component methods became obsolete as of Siebel 7.x.

- Application.MsgBox
- Application.InputBox
- Application.ActiveBusComp
- Application.ActiveApplet
- Application.GotoApplet
- Application.ShowStatus
- Application.ActiveControl
- Application.FindApplet
- BusComp.AllowNotification
- BusComp.SuppressNotification

In many cases, Siebel 7.x implementation alternatives 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, you can get access to the active business component. Additionally, MsgBox and InputBox methods can typically be replaced with some core functions available in JavaScript (through browser script) and some methods introduced in Siebel 7.x.

For example, using browser script, you can use code JavaScript features to provide warnings, dialog boxes, or input boxes through the alert, confirm, and prompt functions. For more information about using JavaScript features available in browser script to replace MsgBox and InputBox, see FAQ 1562 on Oracle’s Siebel SupportWeb.

The RaiseError and RaiseErrorText methods were introduced to provide an alert notification from server script. These methods allow an error message to be displayed as an alert in the Web Client. Lines of code that follow the RaiseError or RaiseErrorText methods are not executed.

For more information about using RaiseError and RaiseErrorText methods, as well as additional information about using browser script and server script in Siebel 7.x, see *Siebel Object Interfaces Reference* and Technical Note 386 on Oracle's Siebel SupportWeb.

#### Related Topics

["About Migrating Siebel 6.x Scripts" on page 281.](#)

["Migrating Siebel Scripts Attached to Controls" on page 285](#)

["Migrating Siebel Scripts Attached to Applets" on page 287](#)

## Migrating Outbound Siebel COM Interfaces

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Client-side application integration in the Siebel Web client can be accomplished by using two methods provided by Jscript.Net, Microsoft's implementation of the ECMA 262 language:

- **ActiveXObject.** Enables and returns a reference to an Automation object
- **GetObject.** Returns a reference to an Automation object from a file

For additional information on the ActiveXObject and GetObject functions, see the Microsoft Web site (<http://www.microsoft.com>) and the Jscript.Net documentation.

The following example illustrates using the ActiveXObject function to access properties and methods of Microsoft Excel, including the Application object and the ActiveSheet.

```
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();
```

In addition to the functions described above, you can use native functions provided by Siebel VB or Siebel eScript to handle client-side application integration:

- Siebel VB
  - CreateObject (to access a client-side automation object)
  - Declare (to access an external DLL)
- Siebel 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 Siebel eScript functions listed above, see *Siebel Tools Online Help*.

#### Related Topics

["About Migrating Siebel 6.x Client-Side Interfaces" on page 283.](#)

["Migrating Inbound Siebel COM Interfaces" on page 292](#)

## Migrating Inbound Siebel COM Interfaces

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Siebel 7.x external applications can interact with the Siebel client through the Web Client Automation Server. The Web Client Automation Server is similar in functionality to the COM Automation Server available in Siebel 6.x, but it does not expose applet or control objects.

The Siebel 7.x Web Client Automation Server enables external applications to interact with the Siebel Web client. The Web Client Automation Server enables external COM-based applications to invoke Siebel business services and manipulate property sets. The Web Client Automation Server is implemented as a COM object resident within the browser.

To enable the Web Client Automation Server, in the [SWE] section of your application's configuration file, make sure that the EnableWebClientAutomation parameter is set to TRUE. When this parameter is set to TRUE, when the application is run and a control is required, an ActiveX control is downloaded to the desktop. This process terminates when the Siebel Web client is terminated normally.

The following example illustrates how a small Microsoft Visual Basic application that uses the Web Client Automation Server and the EAI Siebel Adapter Business Service queries 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 ("Siebel Message")
    status = siebelMsg.SetProperty("IntObjectName", "Sample Account")
    status = siebelMsg.SetProperty("MessageId", "")
    status = siebelMsg.SetProperty("MessageType", "")

    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).GetPropertyCount
```

```
  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
```

For additional information on the Mobile Web Client Automation Server or the Web Client Automation Server, see *Siebel Tools Online Help*.

You might need to modify browser security settings to allow use of the Web Client Automation Server. For security settings and information about deploying ActiveX controls in the browser, see *Siebel System Administration Guide*.

### Related Topics

["About Migrating Siebel 6.x Client-Side Interfaces" on page 283.](#)

["Migrating Outbound Siebel COM Interfaces" on page 291](#)

# Setting Siebel Visibility Modes for Access Control

**Upgrades from:** Siebel 6.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Certain areas of the Siebel 7.x default configuration use Catalog visibility. If you want to continue to use Organization visibility, you must make changes to the business component, view, and picklist properties in order to use Organization visibility. This applies to the following objects:

- Business Components with Catalog Visibility
- Applet picklists with Auto Query Mode or New Query set to None
- Views with Catalog Visibility

After the upgrade, you must carefully evaluate the following business components, views, and picklists, all of which use catalog visibility as the default configuration.

## Business Components that Use Catalog by Default

The following business components use Catalog as the default pop-up visibility type:

- Banter Resolution Item
- Competitor
- Component Product
- Contact Decision Issue
- Decision Issue
- ERM Internal News Category
- 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 by Default

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—Decision Issue
- 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
- Catalog Admin Category Assoc Applet—Training
- Catalog Admin Category Assoc Applet—eEvents Event
- FS Parts & Tools Pick Applet
- FS Use Plan Products Pick Applet
- Incentive Compensation Product Pick Applet
- 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
- Partner Finder List 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
- Service Locator List Applet
- Solution Create List Applet
- Training Course Product Pick 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 that Use Catalog by Default**

The following views use Catalog as the default visibility applet type:

- Auction Place Bid
- Competitive Company Across Catalogs
- Decision Issue View
- Products Across Catalogs
- SHP Sales Product View
- SWLS eChannel Solution Display View
- Sales Tools Across Catalogs
- Service Solution List View (SCW)
- Solutions Across Catalogs
- eAuction Auction Item Search View

### *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, refer to Oracle's Siebel SupportWeb for the *Access Control Upgrade and Migration Guide* technical note.



# 18 Siebel Postmerge Development Tasks

This chapter contains the following topics:

- [“Reviewing Objects Deleted from the Siebel Repository”](#) on page 299
- [“Reviewing Obsolete Objects in the Siebel Repository”](#) on page 300
- [“Migrating Custom Siebel Workflows”](#) on page 301
- [“Upgrading to the Siebel Symbolic String Model”](#) on page 302
- [“Dropping IBM DB2 8-KB Tablespaces and Buffers After a Siebel Merge”](#) on page 303
- [“Updating Siebel Enterprise Application Integration \(EAI\)”](#) on page 303

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Reviewing Objects Deleted from the Siebel Repository

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If an object you have deleted from the Prior Customer Repository exists in the New Customer Repository, the repository merge does not delete the object from the New Customer Repository. After the merge, you must review these objects and verify that they do not adversely affect the operation of the application.

**Prerequisite:** The repository merge must have been successful. See [“Reviewing the Siebel Repository Merge Log Files”](#) on page 235.

### *To generate a list of deleted objects*

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 2 In the Application Upgrades list, select the record of the successful merge.
- 3 Click Query.
- 4 Enter your query criteria in the Object Differences list:

- Click in the In Prior Standard field so that a check mark appears.
  - Click in the Added to New Customized field so that a check mark appears.
  - Click in the In Prior Customized field so that a check mark appears. Then click in it again so that no check mark appears.
- 5 Press Enter to run the query.
- Deleted objects appear in the Object Differences list. You can filter the objects displayed by using the Top Parent Type and Object Type fields.
- 6 Review the list carefully to determine that deleted objects that have been restored to the merged repository will not have an adverse effect on upgraded applications.

### Related Topics

["About the Siebel Repository Merge" on page 89](#)

# Reviewing Obsolete Objects in the Siebel Repository

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Objects that were available in the Prior Siebel Repository but are not available in the New Siebel Repository are obsolete. After performing the repository merge, you can generate a list of obsolete objects using Siebel Tools. Objects that were available in the Prior Siebel Repository are compared with the objects that are available in the New Siebel Repository.

**Prerequisite:** The repository merge must have been successful. See ["Reviewing the Siebel Repository Merge Log Files" on page 235](#).

### *To generate a list of obsolete objects*

- 1 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 2 In the Application Upgrades list, select the record of the successful merge.
- 3 Click Query.
- 4 Enter your query criteria in the Object Differences list:
  - a Click in the In Prior Standard field so that a check mark appears.
  - b Click in the In New Standard field so that a check mark appears. Then click in the field again so that no check mark appears.
  - c Click in the In Prior Customized field so that a check mark appears.

- d Click in the Attribute field so that a check mark appears. Then click in the field again so that no check mark appears.
- 5 Press Enter to run the query.  
All obsolete objects appear in the Object Differences list. You can filter the objects displayed by using the Top Parent Type and Object Type fields.
- 6 Review the list carefully to determine that obsolete objects that have been deleted will not have an adverse effect on upgraded applications.

### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

## Migrating Custom Siebel Workflows

**Upgrades from:** Siebel 7.0.x, 7.5.x, & 7.7.x.

**Environments:** Development environment only.

Custom workflows are not migrated to the New Customer Repository during the upgrep. You must use Siebel Tools to migrate these workflows to the New Customer Repository.

The migration process has the following characteristics:

- All workflows that are in the Prior Customer Repository but not the New Customer Repository are copied to the New Customer Repository. Version 0 workflows are not copied.
- If you have modified a version 0 workflow, change the name of this workflow in the Prior Customer Repository before starting the migration process. This causes the migration process to copy the workflow to the New Customer Repository.
- You can repeat the migration process as needed. You can also exit the migration process before completion and run it again later. You can perform or repeat the migration process at any time after the upgrep and before the upgphys.

### *To migrate custom workflows*

- 1 Start Siebel Tools.  
The Siebel Tools version must be 7.8.x or later.
- 2 Select Tools > Upgrade > Migrate Custom Workflow Processes.  
The Select Repositories dialog box displays.

- 3 Make the following selections in the dialog box, then click OK:

Field Name	Make This Selection
Prior customized repository	<ul style="list-style-type: none"> <li>■ Prior Customer Repository.</li> <li>■ Or select the repository that you are using as the Prior Customer Repository.</li> </ul>
New customized repository	<ul style="list-style-type: none"> <li>■ New Customer Repository.</li> <li>■ Or select the repository you are using as the New Customer Repository.</li> </ul>

The Migrate Workflow Process dialog displays. This dialog displays the steps in the migration. An OK in the first column means the workflow was migrated successfully.

- 4 To stop the migration before it completes, click Cancel.

To restart the migration later, begin at Step 1 above.

- 5 Review the log file for errors.

The log file is located at `Tools\install_dir\log\upgwf.log`.

To locate logged errors, search the log for the word "error."

## Troubleshooting

If the custom workflow migration process encounters an error and cannot migrate a workflow, it prints the word "error" instead of "OK" in the first column of the Migrate Workflow Processes dialog and in the log file.

In most cases, errors are caused by a problem with the workflow record in the Prior Customer Repository. Examine the record, resolve the error, and rerun the workflow migration.

# Upgrading to the Siebel Symbolic String Model

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

Siebel 7.7 introduced a symbolic string model. It replaces the locale-based string model. In the locale-based string model, each UI text string in the Tools repository is part of a UI object definition, such as an applet. The translations of the text string are maintained as child objects in child locale records for each UI object. For commonly used text strings, this means there is a lot of redundancy between UI objects.

The new symbolic string model is object-oriented. A single symbolic string replaces the text string translations. For each language, a text string is defined and assigned to the symbolic string as an attribute. This simplifies multilingual management of text strings throughout the UI.

Some strings will not be converted to the symbolic string model during upgrade. Seed data, error messages, lists of values (LOVs), and non-translatable attributes (such as the text alignment property on a control) will continue to use locale-based strings.

You must execute a conversion utility (consoleapp) to convert and consolidate your custom locale-based strings to the new model. If you plan to install a language pack, it is recommended that you do so before you run the string conversion or consolidation process.

Instructions for converting or consolidating to the symbolic strings model are found in *Using Siebel Tools*.

#### Related Topics

[“About the Siebel Repository Merge” on page 89](#)

[“Upgrade Planning for Siebel String Translation” on page 149](#)

## Dropping IBM DB2 8-KB Tablespaces and Buffers After a Siebel Merge

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Drop the 8-KB tablespace, 8-KB temporary tablespace, and 8-KB bufferpool. Before dropping your 8-KB tablespace, check for the existence of any tables in it by running the following SQL statement:

```
select name from sysibm.systables where TBSPACE='TBS_8K'
```

## Updating Siebel Enterprise Application Integration (EAI)

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

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.

***To upgrade integration objects***

- 1 Determine whether you need to synchronize the integration objects, and synchronize if necessary.

To determine whether you need to synchronize integration objects, review the synchronization considerations in *Integration Platform Technologies: Siebel eBusiness Application Integration Volume II*.

- 2 Validate the integration objects.
- 3 If you receive validation errors, inactivate 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.

# 19 Postupgrade Tasks for the Siebel Database and File System

This chapter contains the following topics:

- [“Reapplying Schema Customizations in the Siebel Database” on page 305](#)
- [“Checking for Inactivated EIM Table Columns in the Siebel Database” on page 308](#)
- [“Validating Dock Objects and Rule Definitions in the Siebel Database” on page 308](#)
- [“Verifying an Upgraded Oracle RDBMS After a Siebel Upgrade” on page 309](#)
- [“Setting Oracle Database Parameters After a Siebel Upgrade” on page 311](#)
- [“Verifying S\\_EVT\\_ACT P1 Index on IBM DB2 After a Siebel Upgrade” on page 311](#)
- [“Updating Siebel File System Attachments” on page 311](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Reapplying Schema Customizations in the Siebel Database

**Upgrades from:** All Siebel releases.

**Environments:** Development environment only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In the current release, tables are obsolete or have been replaced by new tables. If you added extension columns or foreign key (FK) columns to tables that are obsolete in the current release, you must reapply these changes to the new tables.

### Reviewing Obsolete Tables

The upgrade process generates a report that you can review for information about tables that are either obsolete. This report, `xtndobstbl.txt`, lists the following:

- Custom columns in obsolete tables
- Custom foreign key columns pointing to obsolete tables
- EIM mappings for custom foreign key columns to access-control related obsolete tables
- Workflow columns by custom foreign key to obsolete tables
- Customer denormalized columns on obsolete tables

- Obsolete tables in the current release.

Each obsolete table is listed with one of three codes:

- **Not Used.** These tables are not used in the current release, but you may continue to use them. These tables are supported as is (for instance, with docking or EIM).
- **EOL (end of life).** These tables are not used in the current release, and they are not supported in future releases.
- **Inactive.** These tables have been discontinued, and are not supported in the current release. You must move extension columns and foreign key columns that reside on inactive tables to alternate tables.

If no tables are listed in `xtndobstbl.txt`, no action is required.

If this file lists any tables, reapply their custom extensions and foreign key columns to tables in the current release using Siebel Tools. See *Configuring Siebel Business Applications*.

## Obsolete Tables That Cannot Be Used

Table 58 lists examples of previously used tables that are inactive in Siebel 7.7.x (you can no longer use these tables) and lists the suggested new tables to which you need to reapply custom extensions. 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 Oracle’s Siebel Technical Support to validate the reapplication of extension columns and to review the steps necessary to migrate any extension column data to the new tables.

Table 58. Examples of Tables That Are Obsolete in Siebel 7.7.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 created many custom extension columns on the tables `S_EMPLOYEE` or `S_ORG_INT`, both of which are no longer used in Siebel 7.7.x, the joins between the tables are not accurate. This may result in SQL errors when you launch the Siebel client.

In such cases, using Siebel Tools, you must 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 must then review the business component configuration to make sure that the client operates properly. You may need to do this in one of the following cases:

- 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`

## Obsolete Tables You Can Continue To Use

Table 59 lists examples of previously used tables that are obsolete in Siebel 7.7.x, but that you can continue to use.

Table 59. Examples of Tables That Are Not Used in Siebel 7.7.x

Previous Table	Suggested New Table
S_CRSE	S_SRC, S_SRC_EVT (class records) S_PROD_INT, S_PROD_INT_CRSE (course records)
S_CRSE_OFFR	S_SRC, S_SRC_EVT
S_CRSE_REG	S_SRC_EVT_REG
S_CTLG_CAT_REL	S_CTLG_CAT
S_DNB_CONTACT	S_PRSP_CONTACT
S_OPTY_PROD	S_REVN
S_TMSHT_LINE	S_TMSHT_ITEM, S_TMSHT_LN

## Obsolete Tables that Are Now Active

Table 60 lists examples of tables that were unused in previous releases of Siebel Business Applications, but are now used in Siebel 7.7.x.

Table 60. Examples of Previously Unused Tables That Are Used in Siebel 7.7.x

Siebel 7.x Name	Former Name
S_ACT_EMP	S_EVT_ACT
S_ACT_CON	S_EVT_ACT

## Opportunity Product Migration

At Siebel 7.7, data is migrated from S\_OPTY\_PROD to S\_REVN. As a result, if you have custom objects that point to S\_OPTY\_PROD, you must remap the affected base or extension tables. You may have to remap extension columns pointing to S\_OPTY\_PROD as well.

The following base or extension tables are affected in the migration from S\_OPTY\_PROD to S\_REVN:

- S\_FN\_OFFR\_COLT
- S\_FN\_OFFR\_FEE
- S\_FN\_OFFR\_SCHD
- S\_OPTYPRD\_ORG
- S\_OPTY\_PROD1\_FNX
- S\_OPTY\_PROD\_FNX
- S\_OPTY\_PROD\_FNXM

For example, if a child object pointed to S\_OPTY\_PROD, you must manually remap the object to S\_REVN.

## Checking for Inactivated EIM Table Columns in the Siebel Database

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel 7.7, some columns in EIM tables were inactivated (Active = FALSE). This was done to prevent the tables from exceeding the 32-KB table size limitation. In Siebel Tools, query for inactive columns in these tables and verify that this does not affect the operation of the application.

## Validating Dock Objects and Rule Definitions in the Siebel Database

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Changes to visibility rules and dock objects require the assistance of Siebel Technical Support or Oracle’s Siebel Expert Services.

If you deploy Siebel Business Applications to mobile users with local databases, you can run the DI CTUTL utility to verify that all dock objects and rule definitions are correct. Dock objects allow mobile users to synchronize their local databases with the Siebel Server. Rules determine which data users synchronize. For more information about dock objects and rules, see *Siebel Tools Online Help* and *Siebel Remote and Replication Manager Administration Guide*.

### ***To verify that all dock object and rule definitions are correct***

**1** Navigate to the following directory:

Windows: *SI EBEL\_ROOT*\bin

UNIX: *\$SI EBEL\_ROOT*/bin

**2** Type the following command using the parameters specified in [Table 61](#):

```
dictutl /C ODBC_DATASOURCE /U USERNAME /P PASSWORD /D TABLEOWNER /N
"REPOSITORY_NAME" /A y 2> logfile.log
```

Table 61. Command Line Flags for DICTUTL

Flag	Parameter	Description	Required
/C	ODBC_DATASOURCE	ODBC datasource name	Yes
/U	USERNAME	User name to log in to database	Yes
/P	PASSWORD	User password to log in to database	Yes
/D	TABLEOWNER	User name of tableowner	Yes
/N	"REPOSITORY_NAME"	Name of repository for dictionary (the parameter must be bounded within double quotes)	Yes
/A	y or n	Enter the y parameter to ignore the dictionary cache. Enter n if you do not want to ignore the dictionary cache.	Yes

**3** Review the *LOGFILE*.log file:

- a** Open the file in Microsoft Excel. If the file is too large for Excel to display the whole file, break the file into two files and examine each separately.
- b** Query for the word error.
- c** If you locate errors, write down the exact error text.
- d** Query for the word syntax.
- e** If you locate syntax errors, write down the exact error text.
- f** Contact Siebel Technical Support to determine if the errors must be resolved.

The following types of entries indicate that no errors were found:

- ❑ Errors: 0
- ❑ Syntax OK

**Related Topics**

["Preserving Siebel Dock Objects and Visibility Rules" on page 154](#)

# Verifying an Upgraded Oracle RDBMS After a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

You can upgrade your RDBMS before or after upgrading to the current Siebel release.

If you upgrade your Oracle RDBMS after upgrading to the current Siebel release, you must run the Siebel `ddlmp` utility to verify the schema layout of the upgraded database.

This also converts the Siebel database to character-based length for char and varchar data. For additional information about the following procedure, see Alert 1053 on Siebel SupportWeb.

### **To verify the upgraded Oracle RDBMS**

- 1 On the Siebel Server where the Siebel Database Server files are installed, navigate to the following location:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 2 Run the following command:

```
ddlmp /u TableOwner /p TablePassword /c "ODBCDataSource" /f DBSRVR_ROOT/  
DatabasePlatform/schema.ddl /t y /i n /e n /B TableSpace /X IndexSpace /G SSE_ROLE  
/R Y /I SiebelLogDir/ddlctl_verify_RDBMS.log
```

where:

- *TableOwner* is the Siebel table owner name, also called the schema qualifier name.
  - *TablePassword* is the Siebel table owner password.
  - *ODBCDataSource* is the ODBC name for connecting to the database. Enclose the name in quotes.
  - *DBSRVR\_ROOT* is the absolute path to the Siebel Database Server installation directory.
  - *DatabasePlatform* is the Siebel Database Server directory for the Oracle database.
  - *Tablespace* is the Oracle tablespace name for the Siebel database.
  - *IndexSpace* is the Oracle index space name for Siebel database.
  - *SiebelLogdir* is the path to the directory where you want the output log placed (log output directory).
- 3 After the command completes, review the output log files for errors. If the log indicates there are errors, contact Oracle’s Siebel Technical Support.

### **Related Topics**

[“About Upgrading Your RDBMS in the Siebel Environment” on page 121](#)

## Setting Oracle Database Parameters After a Siebel Upgrade

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Databases:** Oracle only.

After the Siebel database upgrade is complete, set the following parameters in `init.ora`:

- **(Oracle 9i only) optimizer\_max\_permutations.** Set this parameter to 100. This parameter is obsolete on Oracle 10g and later.
- For a full list of recommended settings for your postupgrade production environment, see the chapter on configuring the RDBMS in *Siebel Installation Guide* for the operating system you are using.

## Verifying S\_EVT\_ACT P1 Index on IBM DB2 After a Siebel Upgrade

**Upgrades from:** Siebel 6.x.

**Environments:** Development, production test, production.

**Databases:** IBM DB2 only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

After your upgrade, you must verify that the table `S_EVT_ACT` is reorganized on the P1 index. This happens because the append mode flag is set in the repository. If `S_EVT_ACT` is not reorganized on the P1 index, Siebel Remote performance is adversely affected.

## Updating Siebel File System Attachments

**Upgrades from:** Siebel 6.x, 7.0.x, 7.5.x, & 7.7.x.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The following changes have occurred to the structure of the Siebel File System since Siebel 6.x:

- As of Siebel 7.0, Siebel Anywhere requires that file system attachments be located in the `\att` subdirectory of the Siebel File System. If you are upgrading from Siebel 6.x, you must create this subdirectory and copy file system attachments to it.

- At Siebel 7.5, file names of attachments include the attachment table name. If an upgrade migrates the records in an attachment table to a new attachment table, you must run a utility to update the file system attachment names.

Perform the tasks in the following topics to update Siebel File System attachments.

## Updating the Siebel File System Directory

Perform the task in this section if you are upgrading from Siebel 6.x.

You must create an `\att` subdirectory in the Siebel File System and copy file system attachments to it.

### To create the subdirectory in the Siebel File System

- 1 In the existing Siebel File System structure (for example `C:\siebfile`), create an `att` subdirectory if it does not already exist.

Windows example: `C:\siebfile\att`

- 2 Copy (do not move) all files located under the `\siebfile` directory to the `\siebfile\att` directory.

- 3 Verify that files have copied correctly to the `\siebfile\att` directory.

After this has been verified, clean up the file system using the `sfscleanup` utility. See *Siebel System Administration Guide*.

## Updating Attachment File Names

The task in this topic applies to all upgrades. Oracle provides a utility to update attachment file names in the Siebel File system. [Table 62](#) lists the input table names to use when you run the utility. The utility updates all files containing the table names you specify. For example, if you specify the tables `S_OLDTABLE_ATT` and `S_NEWTABLE_ATT`, the utility updates the files system attachments by copying all attachment files containing the string `S_OLDTABLE_ATT` to attachment files containing the string `S_NEWTABLE_ATT`.

Table 62. Input Arguments for the Update Utility

Upgrade Path	Old Table	New Table
Upgrades from: Siebel 6.x.	■ S_EMPLOYEE_ATT	■ S_CONTACT_ATT
Upgrades from: Siebel 6.x.	■ S_ORG_INT_ATT	■ S_ACCNT_ATT

### To update file system attachments

- 1 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

2 Run the following utility:

Windows: `chnge_file_sys.bat OLD_TABLE NEW_TABLE "FILE_SYSTEM"`

UNIX: `chnge_file_sys.ksh -s OLD_TABLE -t NEW_TABLE -f "FILE_SYSTEM"`

where:

- **OLD\_TABLE** = The name of the attachment table in the release you are upgrading from. This table is obsolete in the new release.
- **NEW\_TABLE** = Attachment records in **OLD\_TABLE** were migrated to **NEW\_TABLE** in the new release. The utility copies file system attachments containing the string **OLD\_TABLE** to attachments containing the string **NEW\_TABLE**.
- **"FILE\_SYSTEM"** = name of the directory where the Siebel File System attachments reside (entered inside quotation marks)

Windows example:

```
chnge_file_sys.bat S_EMPLOYEE_ATT S_CONTACT_ATT  
"C:\siebfile\att"
```

UNIX example:

```
chnge_file_sys.ksh -s S_EMPLOYEE_ATT -t S_CONTACT_ATT  
-f "/usr/siebel/siebfile/att"
```

3 Review the renamed files carefully to verify that they can be accessed by Siebel Business Applications.

For example, since **S\_EMPLOYEE\_ATT** is migrated to **S\_CONTACT\_ATT**, verify that files such as **S\_EMPLOYEE\_12-1ABC.SAF** are renamed to **S\_CONTACT\_12-1ABC.SAF**.

## Updating Household Notes and Attachments

The tasks in this section apply to upgrades from Siebel 6.x Financial Services.

If you are upgrading from Siebel 6.x Financial Services, and you are migrating your implementation to the new household design, perform the following procedure to update the file system for household notes and attachments.

The upgrade migrates household note attachments from **S\_ORGGRP\_ATT** to **S\_CONTACT\_ATT**. This updates the records in the database, but does not update file system attachments, which use a naming convention that includes the name of the table they are associated with:

*TableName\_RowID\_Revision#.saf*

You must run two scripts to complete this migration. First, run a script to migrate attachments from **S\_ORGGRP\_ATT** to **S\_CONTACT\_ATT**, and notes from **S\_NOTE\_ORGGROUP** to **S\_NOTE\_CON** in the database. Then, run a script to copy and rename all files named **S\_ORGGRP\*.SAF** to **S\_CONTACT\*.SAF** so that they correspond to new table names.

### To update file attachments

- 1 Run `Mig_HH_note_att.sql` to migrate attachments from **S\_ORGGRP\_ATT** to **S\_CONTACT\_ATT**.

- 2 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 3 Run the following utility:

Windows: `chgng_file_sys.bat OLD_TABLE NEW_TABLE "FILE_SYSTEM"`

UNIX: `chgng_file_sys.ksh -s OLD_TABLE -t NEW_TABLE -f "FILE_SYSTEM"`

where:

- `OLD_TABLE` = The name of the attachment table in the release you are upgrading from. This table is obsolete in the new release.
- `NEW_TABLE` = Attachment records in `OLD_TABLE` were migrated to `NEW_TABLE` in the new release. The utility copies file system attachments containing the string `OLD_TABLE` to attachments containing the string `NEW_TABLE`.
- `"FILE_SYSTEM"` = name of the directory where the Siebel File System attachments reside (entered inside quotation marks)

For example:

```
chgng_file_sys.ksh -s S_ORGGRP_ATT -t S_CONTACT_ATT  
-f "C:\siebfile"
```

- 4 Review the renamed files carefully to verify that they can be accessed by Siebel Financial Services.

## Updating Attachments for Call Reports

The task in this section applies to upgrades from Siebel 6.x.

When you upgrade to Siebel 7.7.x, call report attachments from `S_ACTIVI TY_ATT` are migrated to `S_COMM_ATT`. This data migration updated the records in the database and the pointer to the physical file, but did not update file system attachments, which use a naming convention that includes the name of the table they are associated with: `TableName_RowID_Revision#.saf`.

Therefore, you need to run a script to copy and rename all files named `S_ACTIVI TY*.SAF` to `S_COMM*.SAF` so that they correspond to new table names.

### To generate the file attachment update script

- 1 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 2 Type the following command:

Windows:

```
odbcsql /s "ODBC_DATASOURCE" /u TABLEOWNER  
/p PASSWORD /separator / /a /c rem /e /h  
/o FILESYSTEM_LOCATION/MV_ACT_TO_COMM_ATT_NT.bat  
/I LOGFILE_LOCATION/MV_ACT_TO_COMM_ATT_NT.log SCRIPT_LOCATION/  
MV_ACT_TO_COMM_ATT_NT.sql /v y
```

UNIX:

```
odbcsql /s "ODBC_DATASOURCE" /u TABLEOWNER  
/p PASSWORD /separator / /a /c rem /e /h  
/o FILESYSTEM_LOCATION/mv_act_to_comm_att_unix.ksh  
/I LOGFILE_LOCATION/mv_act_to_comm_att_unix.log SCRIPT_LOCATION/  
mv_act_to_comm_att_unix.sql /v y
```

where:

*"ODBC\_DATASOURCE"* = Data source of the database (entered in quotation marks)

*TABLEOWNER* = Tableowner

*PASSWORD* = Tableowner password

*FILESYSTEM\_LOCATION* = Location of the file system

*LOGFILE\_LOCATION* = Location of the log file

*SCRIPT\_LOCATION* = Location of the script

### **To update the file names of call report attachments**

- Make the file system your current directory, then type the following command:

Windows: MV\_ACT\_TO\_COMM\_ATT\_NT.BAT

UNIX: mv\_act\_to\_comm\_att\_unix.ksh

## **Updating Attachments for S\_LIT**

Perform the task in this topic if you are upgrading from Siebel 7.0.x. At Siebel 7.5, the records in S\_LIT are migrated to S\_CB\_ASSET\_VER. You must update the related file names of attachments from S\_LIT\*SAF to S\_CB\_ASSET\_VER\*.SAF.

### **To update attachments for S\_LIT**

- 1 Navigate to the following directory:

Windows: *SIEBEL\_ROOT*\bin

UNIX: *\$SIEBEL\_ROOT*/bin

2 Enter the following command:

```
Windows: file_upg_mm.bat ODBC_SOURCE USER_NAME PASSWORD TABLE_OWNER  
"FILE_SYSTEM_LOCATION" "SIEBEL_ROOT" "DBSRVR_ROOT"
```

```
UNIX: file_upg_mm.ksh ODBC_SOURCE USER_NAME PASSWORD TABLE_OWNER  
FILE_SYSTEM_LOCATION $SIEBEL_ROOT DBSRVR_ROOT
```

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 tableowner (if your implementation is on iSeries, you can use the value provided for *User\_Name*)
- *FILE\_SYSTEM\_LOCATION* = the directory where the file system resides
- *SIEBEL\_ROOT* = the directory where the Siebel Server is installed
- *DBSRVR\_ROOT* = the directory where Siebel Database Server files are installed

Windows example:

```
file_upg_mm.bat SEBL sadmi n sadmi npw SIEBEL "C:\siebfie" "C:\sea7xx\siebsrvr"  
"C:\sea7xx\dbsrvr"
```

UNIX example:

```
file_upg_mm.ksh SEBL sadmi n sadmi npw SIEBEL /usr/siebel/siebfie $SIEBEL_ROOT  
/usr/siebel/sea7xx/dbsrvr
```

Note that the UNIX syntax does not use quotes around *FILE\_SYSTEM\_LOCATION*, and *DBSRVR\_ROOT*.

3 Review the renamed files carefully to verify that they can be accessed by Siebel Business Applications.

# 20 Postupgrade Tasks for Siebel Applications

This chapter contains the following topics:

- [“Generating Siebel Reporting Relationships” on page 318](#)
- [“Setting Up Siebel Global Time Zone Support” on page 319](#)

## Siebel ERM

- [“Upgrading Siebel ERM Approval Business Process Workflows” on page 320](#)
- [“Upgrading Siebel ERM Customized Microsite and Group News Pages” on page 321](#)
- [“Migrating Course Duration Information for Siebel Training” on page 323](#)
- [“Upgrading the Launch Field in Siebel Training LOV” on page 323](#)
- [“Upgrading the Test Status in Siebel Training” on page 324](#)

## Siebel Marketing

- [“Upgrading Responsibilities in Siebel Marketing” on page 325](#)
- [“Reviewing Siebel Marketing Campaign Data” on page 325](#)
- [“Reviewing Renamed Fields in Siebel Marketing” on page 326](#)
- [“Displaying Regions in Siebel Marketing” on page 326](#)
- [“Revising Program Flowchart Icons in Siebel Marketing” on page 327](#)
- [“Setting Default Campaign Execution Options in Siebel Marketing” on page 328](#)
- [“Upgrading Activity Plans for Programs and Campaigns in Siebel Marketing” on page 328](#)
- [“Upgrading the Newsletter Offer Type in Siebel Marketing” on page 329](#)
- [“Configuring Universal Inbox in Siebel Marketing” on page 329](#)

## Quote and Order Management

- [“Upgrading Siebel Purchase Orders” on page 330](#)
- [“Configuring Siebel Asset-Based Ordering” on page 331](#)
- [“Reviewing Siebel Address Data” on page 332](#)
- [“Configuring Siebel Products and Quotes Data” on page 333](#)

## Workflows

- [“Upgrading Siebel Seeded Workflows” on page 336](#)

- [“Upgrading Inbound Siebel Workflows” on page 336](#)

## Miscellaneous

- [“Removing Siebel Call Center Duplicate Logins” on page 340](#)
- [“Renaming the Siebel Life Sciences Configuration File” on page 341](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

# Generating Siebel Reporting Relationships

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The Generate Reporting Relationships process needs to be executed after the upgrade to Siebel 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). Tables can become out of sync in the following cases:

- After upgrading to Siebel 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).

In Siebel 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 managers and their subordinates are working.
- **Suborganizations view mode.** All Contacts across My Organizations View displays all contacts that are associated to either my organization or any of my organization's suborganizations.

The Generate Reporting Relationships process rebuilds the denormalized relationships in the S\_PARTY\_RPT\_REL table so that the hierarchical view modes 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 re-create the denormalized hierarchical structures in the table. This process generates a large number of transactions for Siebel Remote users and regional nodes.

The standard Siebel 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.

This operation is time and CPU/memory-intensive. The process may take several minutes, depending on the size and complexity of your organizational structures. Do not perform this when you are running other memory-intensive processes.

### *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 use the Mode drop-down that appears in the Web Controls toolbar of Siebel Tools. After you have dragged the button and dropped it 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 using the SRF compiled in the previous step so that the Generate Reporting Relationships button can be invoked.

It is recommended 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. 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 Up Siebel Global Time Zone Support

**Upgrades from:** All Siebel releases.

**Environments:** Development, production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you previously upgraded to Siebel 7.x and you enabled your environment for global time zone, you do not need to repeat the procedure. Instead, you need to upgrade data from columns that were *not* UTC-enabled in Siebel 7.x that are UTC-enabled in the current release.

Global deployments typically span multiple time zones. The global time zone feature converts and stores date and time data using the Universal Time Coordinated (UTC) standard. This feature enables you to track dates and times in a common format across multiple time zones.

Although enabling your environment for global time zone is optional in Siebel 7.x, it is strongly recommended that you operate your production environment with global time zone enabled.

For information on setting up and managing UTC, see *Global Deployment Guide*.

## Upgrading Siebel ERM Approval Business Process Workflows

**Upgrades from:** Siebel 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel 7.5.3, a number of approval business processes were included in the Sample Database. These were intended for use with Employee Self-Service. If you have activated these workflows, you must manually upgrade the approval steps in them.

Also, if you have designed approval business processes for use with Employee Self-Service, you must manually upgrade the approval steps to convert them to business service operations.

The manual upgrade converts the approval steps from a standard workflow operation to a business process operation. This enables the workflow to use new Universal Inbox Business Service functionality.

Both Siebel Marketing and Siebel ERM use Universal Inbox. Application administrators can see all approval tasks across all applications.

### *To manually upgrade approval business process workflows*

- 1 In Siebel Tools Business Process Designer, locate the desired workflow.
- 2 Change the Business Object of the workflow to UInbox Item Task.
- 3 Retrieve the Priority from the workflow's Extract Events Fields step and save the priority in the process property.

The priority is used to set the Inbox Owner Priority.

- 4 Pass the Priority to the RouteInboxItem function:
  - Input Argument: Task.OwnerInfoTaskPriority
  - Type: Process Property
  - Property Name: Item Priority
- 5 In the workflow diagram, locate all steps that check the approver's status.
- 6 Change the step from a Siebel Operation to a Business Service:
  - Type: Business Service
  - Business Service Name: Universal Inbox
  - Business Service Method: GetInboxOwnerInfoEx
  - Leave the Business Component field blank
- 7 Locate the workflow's Deactivate Inbox Owner step.
- 8 Change the step from a Siebel Operation to a Business Service:
  - Type: Business Service
  - Business Service Method: Universal Inbox
  - Business Service Method: DeactivateInboxOwner
  - Pass in InboxItemId, InboxTypeName, and OwnerPartyId to the method
- 9 Deploy and activate the revised workflow.

#### Related Topics

["Configuring Universal Inbox in Siebel Marketing" on page 329.](#)

["Upgrade Planning for Siebel Workflow Designer" on page 147](#)

["Upgrade Planning for Siebel Employee Relationship Management \(ERM\)" on page 140](#)

## Upgrading Siebel ERM Customized Microsite and Group News Pages

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, "How to Perform a Siebel Database Upgrade."](#)

As of Siebel 7.7, virtual business components are provided for microsite and Group News pages. These new business components include important rendering optimizations. If you have created customized page sections, you must change the underlying business component.

These old business components are no longer supported:

- ePortal MM Page Item 1
- ePortal MM Page Item 2
- ePortal MM Page Item 3
- ePortal MM Page Item 7
- ePortal MM Page Item 8

They are replaced by the following virtual business components:

- ERM Microsite Section Body VBC
- ERM Microsite Section Navigation Bar VBC
- ERM Section Page Footer VBC
- ERM Section Page Title VBC
- ERM Microsite Section Quick Picks VBC

### **New Virtual Business Components**

In addition, two virtual business components are provided for creating new page sections:

- For microsite pages: ERM Microsite Section VBC (Copy to create a new Section)
- For Group News pages: ERM Group NewsVBC (Copy to create a new Section)

### ***To upgrade customized microsite or Group News pages***

- 1** In Siebel Tools, locate the applet for the microsite or Group News page that you want to upgrade. Write down the section code from the applet's search specification.
- 2** Locate the appropriate virtual business component:
  - For microsite pages: ERM Microsite Section VBC (Copy to create a new Section)
  - For Group News pages: ERM Group NewsVBC (Copy to create a new Section)
- 3** Copy this virtual business component and give the copy a new name.
- 4** Activate the new virtual business component.
- 5** In the new virtual business component's user properties, activate MicrositeSection. Set it equal to the section code from the applet.
- 6** Find the applet again and change its business component to the new virtual business component.
- 7** Delete the applet's search specification.
- 8** Change the applet's class to CSSSWEFrameListERMPageRender.
- 9** Add the new virtual business component to the views to which the applet belongs.
- 10** Recompile and deploy.

Additional information on upgrading microsite pages and Group News pages can be found in Technical Notes on Oracle's Siebel SupportWeb.

### Related Topics

[“Upgrade Planning for Siebel Employee Relationship Management \(ERM\)” on page 140](#)

## Migrating Course Duration Information for Siebel Training

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

When upgrading from a release prior to Siebel 7.7, course duration information for Siebel Training is not migrated. To display course duration information you must move course duration to an extension column.

### *To migrate course duration information*

- 1 In Siebel Tools, add an extension column to S\_PROD\_INT\_CRSE. Assign the column the same length and data type as S\_SRC\_EVT.DURATION\_DESC.
- 2 Use an SQL command to transfer existing course duration data to the extension column.  
The SQL command should have the following form:  

```
update S_PROD_INT_CRSE a
set a.<name of new extension column> = (select b.DURATION_DESC from S_SRC_EVT b)
where a.par_row = b.row_ID)
```
- 3 Expose the extension column in the user interface.
- 4 (Optional) Create corresponding information in the S\_PROD\_INT\_CRSE.CRSE\_HOUR\_NUM field.

### Related Topics

[“Upgrade Planning for Siebel Employee Relationship Management \(ERM\)” on page 140](#)

## Upgrading the Launch Field in Siebel Training LOV

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel Training, the Distribution Method field is renamed Launch. This field is located in Administration-Training > Training Library.

The list of values for this field was revised in Siebel 7.7. The new values include Download, Launch New Browser, and Launch In-Line.

In previous releases, this LOV contained additional values. If existing records contain these values, you must add these values to the LOV, or the records will not display.

#### Related Topics

[“Upgrade Planning for Siebel Employee Relationship Management \(ERM\)” on page 140](#)

## Upgrading the Test Status in Siebel Training

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, completed tests can have one of two statuses:

- Completed-Passed
- Completed-Failed

In previous releases, completed tests had only one status: Completed.

If you want to revise existing data to reflect the new completion statuses, you can write a script to revise the records. The script should use the following logic:

- If at least one test has reached the maximum attempts and not been passed, mark the status as CompletedFailed.
- If all tests have been passed at least once, mark the status as CompletedPassed. Not all test attempts have to be cleared. If there were three attempts allowed and the student failed on the first two but passed on the third attempt, this is acceptable.
- If at least one test has not been attempted, then there should be no change in the status of the registration. So, if a course had a Course Survey associated to it, and if the user had not attempted it yet, its status cannot go to CompletedPassed.
- Else CompletedFailed takes precedence over CompletedPassed.

#### Related Topics

[“Upgrade Planning for Siebel Employee Relationship Management \(ERM\)” on page 140](#)

# Upgrading Responsibilities in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Siebel 7.7 added many new views. In addition, many views from earlier releases are obsolete. You will need to replace any responsibilities you developed for previous releases. A new set of responsibilities are provided in seed data for all active views in Siebel Marketing.

To implement the new responsibilities, use one of the following approaches:

- Add any custom views you create to the seed data responsibilities for Siebel Marketing.
- Create a set of separate responsibilities for any custom views you create. Provide users with the seed data responsibilities as well as your custom responsibilities.

## Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Reviewing Siebel Marketing Campaign Data

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

At Siebel 7.7, how marketing data is stored was changed. Some Siebel Marketing data is not automatically upgraded:

- Field values from the obsolete Campaign Occurrence header records (from S\_SRC) are not merged to the surviving parent campaign. If you have important historical data that is stored on each campaign occurrence, export the list of obsolete campaign occurrence records from the database and determine which field values (if any) you need to apply to the parent campaigns.
- Exported Lists for campaign occurrences are not re-parented to campaign plans. They remain in the obsolete table S\_DD\_LST\_DISTR.
- Campaign expense records (S\_SRC\_COST) for campaign occurrences are not migrated. This is to prevent campaign expenses from being double-counted after the campaign plan and any campaigns (occurrences) are merged.

During the upgrade, campaign plans and campaigns are merged. This may cause some data to appear double-counted due to re-parenting of similar objects to the same campaign:

- **Program and Campaign Activities.** If Activities are associated to the campaign plan as well as the campaign (occurrence), similar activities may appear twice in the upgraded campaign.
- **Campaign Contacts.** A contact (or prospect) may appear in the same campaign more than once if the campaign member was targeted in multiple occurrences of the same campaign plan. After upgrade, this campaign member will appear in more than one campaign load for the same campaign.

### Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

## Reviewing Renamed Fields in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, some Marketing applet fields have been renamed. If you have written any customized user documentation, you need to update references to these fields. Renamed fields are shown in [Table 63 on page 326](#).

Table 63. Renamed Fields in Siebel Marketing

Business Component Field Name	Business Component	Original Caption	New Caption
Budget	Program Container	Budget	Assigned Budget
Period	Campaign	Period	Execution Period
Response Type	Campaign	Response Type	Enabled Follow Up Action

### Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

## Displaying Regions in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, Region is associated with the Region hierarchy rather than the Region LOV field. For this reason, after the upgrade the Region field for Marketing Plans, Programs, and Campaigns does not display.

### ***To display marketing regions***

- 1 Create new Marketing Regions under Administration – Location to correlate to the previous LOV values for the Region field.

For more information, see the *Siebel Marketing Installation and Administration Guide*.

- 2 Rename the Region LOV field and add it to the applets in the user interface.

This provides backward compatibility.

### **Related Topics**

[“Upgrade Planning for Siebel Marketing” on page 141](#)

## **Revising Program Flowchart Icons in Siebel Marketing**

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

After an upgrade from releases prior to Siebel 7.7, the icons in the Marketing Program Flowchart view display larger than normal.

### ***To revise the size of Program Flowchart icons***

- 1 Right-click in the Program Flowchart applet and deselect Snap to Grid.
- 2 Select the desired icon and reduce its size.

For more information on setting up Program Flowcharts, see the *Siebel Marketing Installation and Administration Guide*.

### **Related Topics**

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Setting Default Campaign Execution Options in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#) It is for upgrades from releases prior to Siebel 7.7. It is not mandatory but may be required by local business processes.

Campaign Execution options are important for controlling the load behavior, launch behavior, assignment behavior, and collaboration options for campaigns. Each Campaign Execution option has an assigned default value. Review your marketing business process to confirm that each Campaign Execution Option is set correctly.

For a discussion of how to set default campaign execution options, see the *Siebel Marketing Installation and Administration Guide*.

## Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Upgrading Activity Plans for Programs and Campaigns in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This postupgrade task is not mandatory but may be required by local business processes.

As of Siebel 7.7, the Activity Template type DBM Campaign is obsolete and is not used. Change the template type to Campaign.

## To upgrade Activity Plans

- 1 Navigate to Administration--Data > Activity Templates.
- 2 Query for templates of type DBM Campaign.
- 3 For each template, update the type to Campaign.

## Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Upgrading the Newsletter Offer Type in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This postupgrade task is not mandatory but may be required by local business processes.

As of Siebel 7.7, the offer type eNewsletter is obsolete. If you have existing offers that are of type eNewsletter, revise the type to Email.

## *To upgrade the offer type*

- 1 Navigate to Offers > All Offers.
- 2 Query for Channel = eNewsletter.
- 3 For each offer, change the offer type to Email.

## Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Configuring Universal Inbox in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

As of Siebel 7.7, Universal Inbox is implemented as a business service. It provides management of approval queues in both Siebel ERM and in Siebel Marketing. Application administrators can see all the queued tasks in Universal Inbox across all applications.

This means that the Siebel ERM administrator can see tasks queued in Siebel Marketing and vice versa. If this cross-application visibility violates data visibility policies at your site, create a copy of the Universal Inbox. Then assign the copy to the workflows in one of the applications. Use the preconfigured version of Universal Inbox in the other application. This creates physically separate approval queues.

## Universal Inbox in Siebel Marketing

As of Siebel 7.7, Siebel Marketing uses a new business service, Universal Inbox. It provides centralized management of approval queues and is used by both Siebel Marketing and Siebel ERM. Application administrators can see all approval tasks across all applications.

### Related Topics

[“Upgrade Planning for Siebel Marketing” on page 141](#)

# Upgrading Siebel Purchase Orders

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If upgrading from a release prior to Siebel 7.7, you must 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.

### ***To update the Transaction Amount field in Payment Lines***

- Enter the following command:

Windows: `SIEBEL_ROOT\bin\pmntupgd /u USERNAME /p PASSWORD /l LANG /c CFG_FILE /d DATA_SOURCE`

UNIX: `$SIEBEL_ROOT/bin/pmntupgd /u USERNAME /p PASSWORD /l LANG /c CFG_FILE /d DATA_SOURCE`

Where

- `USERNAME` is the Siebel user login name
- `PASSWORD` is the Siebel login password
- `LANG` is the language used
- `CFG_FILE` is the configuration file used to launch the application
- `DATA_SOURCE` is the data source used from the .cfg file

# Configuring Siebel Asset-Based Ordering

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Production test, production.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Siebel 7.7 introduced several architectural changes to the asset-based ordering workflows and views:

- **Asset-Based Ordering workflows.** The asset-based ordering workflows have been modified. If you are upgrading from Siebel 7.0.4 or Siebel 7.5, and you have modified these workflows, you must reapply your changes to these workflows after the upgrade.
- **Quote > Orders view.** The Sales Order button now uses a named method to invoke the SIS OM Quote To Order Workflow--PMT Version. You can now pass arguments to the workflow. In Siebel 7.5, this button was hard-coded. If you have written scripts or made other changes regarding the Sales Order button, consider modifying them to take advantage of the named method.
- **Order Header applet.** The Submit button now uses a named method to invoke the SIS OM Submit Order Process. You can now pass arguments to the workflow. In Siebel 7.5, this button was hard-coded. If you have written scripts or made other changes regarding the Submit button, consider modifying them to take advantage of the named method.
- **Disconnect and Modify workflows.** The named methods used to invoke the Disconnect and Modify workflows now pass the additional properties shown in [Table 64](#). If you have written scripts or made other changes regarding these buttons or named methods, you must manually add these after the upgrade.

Table 64. Additional Properties Passed by Named Methods

Property	Explanation
"Due Date", "Today() + 1"	Sets the Due Date on all line items.
"BC Context", "Asset"	Passed as a property to the workflow.
"Compound Product Number", "[Compound Product Number]"	The Compound Product Number is used for Network Order Entry.

- **Delta-In-Place feature.** Delta-In-Place updates the action code of a line item when a specific line item field value is updated. For example, if a user updates the Service ID value of a product in the Order Line Item applet, the action code changes from - (dash) to Update.

In Siebel 7.5, the Delta-In-Place feature was implemented through User Properties on the Quote Item and Order Entry – Line Items business components.

For example:

SIS OM On Field Update Set 6

```
"Service Id", "Action Code", "If ([Action Code] = LookupValue('DELTA_ACTION_CODE', 'Existing'), LookupValue('DELTA_ACTION_CODE', 'Modified'), [Action Code])"
```

In Siebel 7.7, the Delta-In-Place feature was moved to the SIS OM PMT Service workflow. The SIS OM PMT Delta method that is used for changes made to products within the Configurator session is now used to generate the action code for changes to line-item level fields.

If you have added new fields to the Order Management process (quote, order, asset) and then used the SIS OM Field Update Set user property, you must create new user properties in the SIS OM PMT Service to replace these.

For example, if you created a new custom field called Point of Presence, you can delete the old SIS OM On Field Update Set user properties and create a new one as follows:

- **Name:** Delta Line Item Compare Field 26
- **Value:** [Point Of Presence]:[Point of Presence]
- **Siebel Configurator runtime: Done button.** In Siebel 7.0.4 and 7.5, when a user clicks Done in a Siebel Configurator session, Siebel EAI transfers user selections into Quote and Order line items. This process used Quotes, Orders, and Assets-related business components.

As of Siebel 7.7, this process uses new, smaller business components. This improves performance. The new business components begin with "MACD." If you have modified the original business components, you must apply these changes to the MACD business components.

## Reviewing Siebel Address Data

### Upgrades:

- From Siebel Industry Applications (SIA) 6.x, (except Communications, Media, and Energy) to Siebel SIA 7.7.x
- From Siebel Financial Services 6.x and 7.0.x to Siebel SIA 7.7.x
- From Siebel Business Applications 6.x and 7.x to Siebel Business Applications 7.7.x as part of upgrading to Siebel SIA 7.7.x
- From Siebel Financial Services 6.2.1 on IBM z/OS platforms to Siebel SIA 7.7.x on IBM z/OS platforms

**NOTE:** This topic does not apply to Siebel Business Applications (HOR) that you are upgrading to a later release of Siebel Business Applications (HOR).

**Environments:** Production test, production.

**Databases:** All databases.

**Platforms:** MS Windows, UNIX, IBM z/OS.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The way address data is stored changed in Siebel 7.7. The upgrade migrated data from the table S\_ADDR\_ORG to S\_ADDR\_PER. The upgrade uses the following method for preserving address data:

- If multiple records have the same value for ADDR\_NAME within or across S\_ADDR\_ORG and S\_ADDR\_PER, the upgrade process preserves ADDR\_NAME for one of the records and appends the ROW\_ID to ADDR\_NAME in the others.
- The upgrade process truncates the value of ADDR\_NAME to fit the size of the column.
- The upgrade generates a report listing the records with a duplicate ADDR\_NAME within and across S\_ADDR\_ORG and S\_ADDR\_PER.

After the upgrade completes, review this report and edit or delete records from S\_ADDR\_PER as desired.

**NOTE:** This report is also generated for Siebel 7.0.x Siebel Industry Solutions upgrades to Siebel 7.7 Siebel Industry Applications. Ignore this report for these upgrades.

### *To review address records after upgrade*

- 1 Review the report generated by the upgrade:  
Windows: *SIEBEL\_ROOT\log\rpt\_dup\_addr\_names.txt*  
UNIX: *\$SIEBEL\_ROOT/1og/rpt\_dup\_addr\_names.txt*
- 2 Use the report to identify records that are duplicates.
- 3 Use the application or EIM to delete or revise records as needed.

### **Related Topics**

[“Upgrade Planning for Migrating Siebel Address Data” on page 139](#)

# Configuring Siebel Products and Quotes Data

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

You may need to adjust or configure some customizable products and quotes after upgrade.

## Customizable Products

After upgrading to Siebel 7.7.x, the administrator must verify that customizable products work as needed. In order to do so, the administrator must validate and release customizable products for use. This action is performed in the Product Administration screen. If the administrator does not release customizable products after the upgrade, customizable products are not active.

Packaged products do not require additional modifications in the postupgrade process.

## Quote Items

After upgrade, run the following script to fix an issue where quote items had trailing spaces added accidentally. Running 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
;
```

## File System Attachments for Quotes

During the upgrade, quote attachments from S\_QUOTE\_ATT are migrated to S\_ORDER\_ATT. This updated the records in the database and the pointer to the physical file, but did not update file system attachments, which use a naming convention that includes the name of the table they are associated with: *TableName\_RowID\_Revision#.saf*.

You must run a script to copy and rename certain files named S\_QUOTE\*.SAF to S\_ORDER\*.SAF so that they correspond to new table names.

### *To update the file names of quote attachments*

- 1 Navigate to the following directory:

Windows: *SIEBEL\_ROOT*\bin

UNIX: *\$SIEBEL\_ROOT*/bin

- 2 Type the following command:

Windows:

```
odbcsql /s "ODBC_DATASOURCE" /u TABLEOWNER
/p PASSWORD /separator / /a /c rem /e /h
/o FILESYSTEM_LOCATION/MV_QUOTE_TO_ORDER_ATT_NT.bat
/l LOGFILE_LOCATION/MV_QUOTE_TO_ORDER_ATT_NT.log SCRIPT_LOCATION/
MV_QUOTE_TO_ORDER_ATT_NT.sql /v y
```

UNIX:

```
odbcsql /s "ODBC_DATASOURCE" /u TABLEOWNER
/p PASSWORD /separator / /a /c rem /e /h
/o FILESYSTEM_LOCATION/mv_quote_to_order_att_unix.ksh
/l LOGFILE_LOCATION/mv_quote_to_order_att_unix.log SCRIPT_LOCATION/
mv_quote_to_order_att_unix.sql /v y
```

where:

*"ODBC\_DATASOURCE"* = Data source of the database (entered in quotation marks)

*TABLEOWNER* = Tableowner

*PASSWORD* = Tableowner password

*FILESYSTEM\_LOCATION* = Location of the file system

*LOGFILE\_LOCATION* = Location of the log file

*SCRIPT\_LOCATION* = Location of the script

Flag	Parameter	Description	Required
/s	<i>"ODBC_DATASOURCE"</i>	Data source of the database (entered in quotation marks)	Y
/u	<i>TABLEOWNER</i>	User name to log into database	Y
/p	<i>PASSWORD</i>	Password to log into database	Y
/a	Not applicable	Turn on ODBC auto-commit for session	N
/c	<i>rem</i>	Remark. Specify the string that begins at comment (at the beginning of the line)	Y
/o	<i>FILESYSTEM_LOCATION</i>	Directory where the file system resides	Y
/l	<i>LOGFILE_LOCATION</i>	Write errors and status to log file specified here	Y
	<i>SCRIPT_LOCATION</i>	Location of the script	Y
/v	Not applicable	Turn on statement printing at execute	N

- 3 Change your directory to the file system, then type the following command:

Windows: MV\_QUOTE\_TO\_ORDER\_ATT\_NT.BAT

UNIX: mv\_quote\_to\_order\_att\_unix.ksh

## Upgrading Siebel Seeded Workflows

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Environments:** Development, production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

In Siebel 7.7, workflow definitions were relocated to the Tools Repository.

Customizations to seeded workflows were saved and migrated during upgrade, but you must manually reimplement them in order for them to work properly.

### To upgrade seeded workflows

- 1 Purge workflow instances.

For information on purging these tables, see the topic on purging workflow process instances in *Siebel Business Process Designer Administration Guide*.

- 2 In the Siebel Repository, revise each seeded workflow so that a new copy is created with a new version number.
- 3 Manually merge in your customizations, and then deploy and activate the workflow.

## Upgrading Inbound Siebel Workflows

**Upgrades from:** Siebel 6.x & 7.0.x.

**Environments:** Development, production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Change inbound workflows that contain a “String” type process property to pass the value into type Binary; otherwise, the workflow presents 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 Siebel 6.x, manually change the name of the “EAI MQSeries Transport” business service to the name “EAI MQ Series Server Transport”; otherwise, the workflow presents 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.
```

### Related Topics

[“Upgrade Planning for Siebel Workflow Designer” on page 147](#)

# Universal Customer Master (UCM) Upgrade Steps After Repository Merge

**Upgrades from:** UCM 7.8.2 to 8.0

**Environments:** Development environment only.

Siebel UCM 7.8.2 is shipped with .sif files in addition to the standard 7.8.2 repository. In the Siebel 7.8.2 to 8.0 upgrade process, the Prior Standard Repository is a standard Siebel 7.8.2 repository which does not contain the UCM 7.8.2 additions gained by applying the UCM .sif file. This inconsistency may cause some false conflicts and the 8.0 changes may be overridden by the repository merge conflict resolution process. Some of the mis-picked values may cause functional impact in various UCM areas, such as UI, source data and the history, batch manager and so on. To retain out of the box 8.0 functionality for UCM 8.0, you need to perform the following procedure.

## *To change the Resolution field in the Application Upgrade Attribute list*

- 1 Log in to Siebel Tools following your repository merge.
- 2 From the application menu, click Screens > Application Upgrader > Application Upgrade Attribute List.
- 3 In the Attribute Differences list, find the objects listed in [Table 65](#) and change the value in the Resolution field from "Use Custom Value" to "Use Standard Value."

Table 65. UCM Objects prior to Resolution field value change

Object Name	Top Parent Name	Top Parent Type	Object	Attribute	Prior Customized Value
UCM Account Address Source Data and History	UCM Account Address Source Data and History	Business Component	Business Component 'UCM Account Address Source Data and History'	Search Specification	[Object Name] = "CUT Address"
UCM Contact Address Source Data and History	UCM Contact Address Source Data and History	Business Component	Business Component 'UCM Contact Address Source Data and History'	Search Specification	[Object Name] = "Personal Address"
UCM Account History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Account History Flat List View' of Screen 'CIF Administration Screen'	Menu Text - String Reference	SBL_ACCOUNT_SOURCE_DATA_UCM

Table 65. UCM Objects prior to Resolution field value change

Object Name	Top Parent Name	Top Parent Type	Object	Attribute	Prior Customized Value
UCM Account History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Account History Flat List View' of Screen 'CIF Administration Screen'	Viewbar Text - String Reference	SBL_ACCOUNT_SOURCE_DATA_UCM
UCM Contact History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Contact History Flat List View' of Screen 'CIF Administration Screen'	Menu Text - String Reference	SBL_CONTACT_SOURCE_DATA_UCM
UCM Contact History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Contact History Flat List View' of Screen 'CIF Administration Screen'	Viewbar Text - String Reference	SBL_CONTACT_SOURCE_DATA_UCM
Yes	UCM Batch Process: 3	Workflow Process	WF Step Branch 'Yes' of WF Step 'Last Page?' of Workflow Process 'UCM Batch Process: 3'	Expression	(LastPage All Must Match (Ignore Case)('true'))

- 4 After changing the Resolution field value from “Use Custom Value” to “Use Standard Value” the objects will reflect the values displayed in [Table 66](#).

Table 66. UCM Objects after Resolution field value change

Object Name	Top Parent Name	Top Parent Type	Object	Attribute	New Standard Value
UCM Account Address Source Data and History	UCM Account Address Source Data and History	Business Component	Business Component 'UCM Account Address Source Data and History'	Search Specification	[Parent Type] = 'Account'
UCM Contact Address Source Data and History	UCM Contact Address Source Data and History	Business Component	Business Component 'UCM Contact Address Source Data and History'	Search Specification	[Parent Type] = 'Contact'
UCM Account History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Account History Flat List View' of Screen 'CIF Administration Screen'	Menu Text - String Reference	SBL_ACCOUNTS-1004224419-01T
UCM Account History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Account History Flat List View' of Screen 'CIF Administration Screen'	Viewbar Text - String Reference	SBL_ACCOUNTS-1004224419-01T
UCM Contact History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Contact History Flat List View' of Screen 'CIF Administration Screen'	Menu Text - String Reference	SBL_CONTACTS-1004224924-40K

Table 66. UCM Objects after Resolution field value change

Object Name	Top Parent Name	Top Parent Type	Object	Attribute	New Standard Value
UCM Contact History Flat List View	CIF Administration Screen	Screen	Screen View 'UCM Contact History Flat List View' of Screen 'CIF Administration Screen'	Viewbar Text - String Reference	SBL_CONTACTS-1004224924-40K
Yes	UCM Batch Process: 3	Workflow Process	WF Step Branch 'Yes' of WF Step 'Last Page?' of Workflow Process 'UCM Batch Process: 3'	Expression	

## Removing Siebel Call Center Duplicate Logins

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

During the upgrade to Siebel 7.7.x, employee and contact data were moved to the same tables. As a result, your database may contain duplicate logins for Contact user login names.

The upgrade appends +ROW\_ID to duplicate logins. You must resolve user logins after the upgrade; otherwise, users are unable to log in. For example, contact user login names 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 brings up all names that are appended with “+ROW\_ID”.
- 3 Repeat this procedure for each User Administration view (Employees, Persons, and Users).

# Renaming the Siebel Life Sciences Configuration File

**Upgrades from:** Siebel 6.x.

**Environments:** Production test, production.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

If you are upgrading from Siebel 6.3 Life Sciences to Siebel 7.7.x, you must rename your .cfg file before you upgrade your mobile and connected clients as instructed in *Siebel Anywhere Administration Guide*. This action prevents an error from occurring when you log in to the ePharma client.

After you install the upgraded Client Executables and Client Executables ENU, but before you create the packager kit with Siebel Anywhere, copy the file named ePharma.cfg and rename the copy to Pharma.cfg in the new build client before you create the packager kit.

If you do not rename the .cfg file, you may receive the following error message when you try to log in to the upgraded ePharma client: Unable to load configuration file  
“D:\63\client\Bin\ENU\Pharma.cfg”.



# 21 Tuning the Siebel Upgrade Files

This chapter contains the following topics:

- [“Starting and Stopping Siebel Upgrade Tuner” on page 343](#)
- [“Managing Parallel Threads Using Siebel Upgrade Tuner” on page 345](#)
- [“Managing Zero-Row SQL Commands Using Siebel Upgrade Tuner” on page 348](#)
- [“Transferring UNIX Files for Use by Siebel Upgrade Tuner” on page 351](#)
- [“Rolling Back Siebel Upgrade Tuner Changes” on page 354](#)

## Related Topics

[Chapter 4, “How the Siebel Database Upgrade Works”](#)

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Starting and Stopping Siebel Upgrade Tuner

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

This topic describes how to start the Upgrade Tuner (upgtuner) from the Siebel Server installation directory on a Windows host. It also describe how to save or discard your changes when exiting Upgrade Tuner.

Use the Upgrade Tuner to improve the performance of table creation, index creation, and SQL execution during the production upprep step in the production test environment.

## Prerequisites

- In the production test environment, you must have completed the production upprep step.
- You must have run the logparse utility on the production upprep log files. See [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219](#).
- For UNIX platforms, you must have transferred files to a Windows host on which a Siebel Server is installed. See [“Transferring UNIX Files for Use by Siebel Upgrade Tuner” on page 351](#).

## Starting Upgrade Tuner

Use this procedure to start Upgrade Tuner.

### *To start Upgrade Tuner*

- Enter the following command:

```
SIEBEL_ROOT\bin\upgtuner /c LOGPARSE_SUMMARY_LOCATION
```

where:

*LOGPARSE\_SUMMARY\_LOCATION* = location of summary.xml generated by the log parse utility for the production upgrep.

For example:

```
upgtuner /c SIEBEL_ROOT\log\upgrep_prod_77\summary.xml
```

The Upgrade Tuner Process Information page appears.

Alternative: Start Upgrade Tuner without the /c option. Click OK at the error pop-up. When Upgrade Tuner displays, click Browse and navigate to the summary.xml file.

## Saving Your Changes and Exiting Upgrade Tuner

Use this procedure to save your changes and exit Upgrade Tuner.

### *To save your changes and exit Upgrade Tuner*

- 1 Evaluate the changes you have made and revise them as needed.  
Upgrade Tuner does not revise the upgrade files until you exit.
- 2 Click Save+Exit.
- 3 Click Yes in the pop-up window that asks you to confirm if you want to save and exit.  
Upgrade Tuner applies your changes to the upgrade files and exits.

## Discarding Your Session Changes and Exiting Upgrade Tuner

Use this procedure to discard the changes you have made in the current session and exit Upgrade Tuner. No changes are made to upgrade files.

### Parallelize Table or Index Creation Pages

Changes are discarded and do not display the next time you start Upgrade Tuner, if:

- You moved tables or indexes between threads, these changes are discarded.
- You created new threads, the threads are discarded.

### Deactivate 0-Row SQLs

Changes are discarded and do not display the next time you start Upgrade Tuner, if:

- You inactivated a command, this change is discarded. The next time you start Upgrade Tuner, no check mark displays in the Inactive column. The command remains active in the SQL file.
- You activated a command, this change is discarded. The next time you start Upgrade Tuner, a check mark displays in the Inactive column, and the command remains inactive in the SQL file.

### *To discard your changes and exit Upgrade Tuner*

- 1 Click Cancel.
- 2 Click Yes in the pop-up window that asks if you to confirm you want to discard your changes and exit.

### Related Topics

[“About Tuning Siebel Production Upgrade Files” on page 109](#)

# Managing Parallel Threads Using Siebel Upgrade Tuner

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only.

**Databases:** Oracle only.

Upgrade Tuner allows you to create, edit, and delete parallel thread for table and index creation. This improves upgrade performance by reducing the amount of time required to complete table and index creation.

You create, edit, and delete threads in the Parallelize Table Creation page and the Parallelize Index Creation page. These two pages have the same layout.

### Prerequisites

- In the production test environment, you must have completed the production upgrep step.
- You must have run the Logparse utility on the production upgrep log files. See [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219](#).
- For UNIX platforms, you must have transferred files to a Windows host on which a Siebel Server is installed. See [“Transferring UNIX Files for Use by Siebel Upgrade Tuner” on page 351](#).
- Start Upgrade Tuner. See [“Starting and Stopping Siebel Upgrade Tuner” on page 343](#).

## Displaying Threads

You can view and sort the contents of threads or view items sorted across threads:

- **Default sort.** The default sort is all the items in the Serial Thread sorted from highest to lowest cost. The default sort displays when you start Upgrade Tuner, add a thread, or remove a thread. To reverse the sort order, click on the Serial Thread column head.
- **Contents of a thread.** Click on the column head for that thread. To reverse the sort-order, click on the column head again.
- **All items in all threads sorted by cost.** Click on the Cost per Table column head. To reverse the sort-order, click on the column head again.

## Creating Parallel Threads

Use this procedure to create a parallel thread for table creation or index creation. Upgrade Tuner automatically names threads Parallel Thread 1, Parallel Thread 2, and so on. You cannot edit thread names.

You start creating parallel threads by creating Parallel Thread 1 and Parallel Thread 2 together. You must assign at least one table or index to each of these threads.

All threads you create must contain at least one table or index.

### *To create Parallel Thread 1 and Parallel Thread 2*

- 1 In the Serial Thread, select a table or index, and move it to the right using the arrow key on the keyboard.

Upgrade Tuner creates Parallel Thread 1 and Parallel Thread 2. It then assigns the table or index to Parallel Thread 1.

- 2 Move at least one table or index to Parallel Thread 2, to populate both threads.

### *To create additional parallel threads*

- Select a table or index and move it to the right using the arrow key on the keyboard.

When you move the table or index to the highest-numbered thread and click the arrow key again, Upgrade Tuner creates a new thread and places the table or index in the new thread.

**TIP:** Another way to create a new thread is to right-click in a row. In the drop-down menu, select the last thread listed.

## Moving Items Between Threads

Use the left and right arrow keys to move tables or indexes between threads, including the Serial Thread.

**NOTE:** You cannot save and exit if any thread is empty.

## Deleting a Thread

Use this procedure to delete an existing thread. You cannot delete the Serial Thread.

You must delete Parallel Thread 1 and Parallel Thread 2 together. You must delete all other threads before deleting Parallel Thread 1 and Parallel Thread 2.

### *To delete a thread other than Parallel Thread 1 or Parallel Thread 2*

- 1 Click the column head for the desired thread.  
This action sorts the list so that all items in that thread appear at the top.
- 2 Right-Click in the column head, and choose “Move all items to the serial thread” from the pop-up menu.
- 3 Right-Click in the column head, and choose “Remove thread” from the pop-up menu.  
Upgrade Tuner deletes the thread and renames all higher-numbered threads.

### *To delete Parallel Thread 1 and Parallel Thread 2*

- 1 Right-Click the column head for Parallel Thread 2, and choose “Move all items to the serial thread” from the pop-up menu.
- 2 Right-Click the column head for Parallel Thread 1, and choose “Move all items to the serial thread” from the pop-up menu.
- 3 Right-Click in the column head for Parallel Thread 2, and choose “Remove thread” from the pop-up menu.  
Upgrade Tuner deletes Parallel Thread 1 and Parallel Thread 2.

## Evaluating Upgrade Performance Improvement

To evaluate production upgrade performance improvement, use the two fields at the top of the page:

- **Total cost of sequential table (or index) creation.** Displays the time to create tables or indexes when no parallel threads are used.
- **Total cost of parallelized table (or index) creation.** Displays the time to complete the upgrade using the parallel threads you have created. The time is computed by adding the Serial Thread time and the longest-running parallel thread time.

The difference between the sequential creation time and the parallelized creation time is an estimate of the reduction in upgrade time from using parallel threads.

You can reduce upgrade time further by performing the following actions:

- Move additional items from the Serial Thread to a parallel thread
- Move items from the longest-running parallel thread to other threads or a new thread

The goal is to reduce both the Serial Thread time and longest-running parallel thread time to a minimum. Because each new parallel thread requires additional memory and CPU cycles, you may need to experiment with the number of parallel threads to optimize upgrade performance.

## Related Topics

[“About Tuning Siebel Production Upgrade Files” on page 109](#)

# Managing Zero-Row SQL Commands Using Siebel Upgrade Tuner

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

The upgrade scripts support all the tables in the Siebel data model. This support means the tables may contain SQL commands that run against tables that are not included in your Siebel database, that are empty, or do not contain data that applies to a specific SQL command. By inactivating such SQL commands, you can reduce the time required to perform the production upprep.

The Deactivate 0-Row SQLs page displays a list of SQL files that contain commands that returned zero rows. This means the command does not affect any data and does not change the database schema. The screen displays only upgrade commands executed natively by the RDBMS. The screen does not display SQL commands executed using odbcsql.

The SQL files are located in `DBSRVR_ROOT\DBPLATFORM\upgrade\VERSION\`, for example, `db2udb\upgrade\V7_7\`.

When you select a file, the command that returned zero rows displays in the lower half of the screen. You can then either deactivate or activate the command. You cannot edit the displayed command.

When you deactivate a command and save your changes, Upgrade Tuner opens the SQL file containing the command and inserts (Execute=No) in the command. When you activate a command, Upgrade Tuner removes (Execute=No) from the command.

## Prerequisites

- In the production test environment, you must have completed the production upprep step.
- You must have run the logparse utility on the production upprep log files. See [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219](#).
- For UNIX platforms, you must have transferred files to a Windows host on which a Siebel Server is installed. See [“Transferring UNIX Files for Use by Siebel Upgrade Tuner” on page 351](#).
- Analyze your customizations and the nature of application data. Verify that you understand the role of any new tables you have added.
- Start Upgrade Tuner. See [“Starting and Stopping Siebel Upgrade Tuner” on page 343](#).

## Displaying Zero-Row SQLs

You can view and sort zero-row SQLs in several ways:

- **Default sort.** The default sort order is the order in which the zero-row commands appear in the driver files. Any inactivated SQL commands, including those inactivated in previous sessions appear at the end of the list. The default sort order displays when you start Upgrade Tuner.
- **Display items sorted by cost.** To sort commands from longest-running time to shortest, click the Net Cost column head. To reverse the sort order, click Net Cost again. Commands inactivated prior to this session appear at the end of the list.
- **Display commands activated or deactivated in the current session.** Click the Inactive column head. Items display at the beginning of the list. The word Changed displays in the Inactive column for these items. Items that have been deactivated display check marks. Items that have been activated do not.
- **Display commands inactivated in previous sessions.** Click the Net Cost column head and scroll to the end of the list. Inactivated commands do not have a check mark in the Inactive column and do not display the word Changed.
- **Display commands activated in previous sessions.** The display of SQL commands does not provide a way to identify commands activated in a previous Upgrade Tuner session. When you activate a command, write down its SQL file name and SQL tag number so you can locate the command in future sessions.
- **Display all the zero-row SQL commands in a file.** Click on the SQL File column head. This action sorts the file names alphabetically. To reverse the sort order, click on the column head again.

## Deactivating Zero-Row SQL Commands

Use this procedure to deactivate SQL commands that do not affect any data.

### *To deactivate zero-row SQL commands*

- 1 Click the Deactivate 0-Row SQLs tab in Upgrade Tuner.  
The Deactivate 0-Row SQLs screen appears.
- 2 Click the Net Cost column head.  
This sorts the entries so that the longest running SQL commands appear first. If they do not, click the column head again. Commands deactivated in previous Upgrade Tuner sessions display at the end of the list.
- 3 Click in a row to display a command that returned zero rows.
- 4 Carefully evaluate whether you need this command for your upgrade.
- 5 Write down the net cost of the command.  
You can use a spreadsheet to keep track of net cost changes, if you prefer.
- 6 To deactivate the command, click in the check box in the Inactive column.  
The following occurs:

- A check mark displays indicating the command is inactive.
- The word Changed appears next to the check mark to indicate the change was made in this session.
- The time displayed in the Net Cost column changes to N/A.
- When you save and exit, Upgrade Tuner inactivates the command in the SQL file.
- The next time you start Upgrade Tuner, a check mark displays in the Inactive column for the command, but the word Changed does not.

## Activating Zero-Row SQL Commands

Use this procedure to activate SQL statements that do not affect any data.

### *To activate zero-row SQL commands*

- 1 Click the Net Cost column head, and then scroll to the end of the list.

This sorts commands by running time. Inactive commands have a running time of N/A and always appear at the end of the list.

- 2 Click in a row to display a command that returned zero rows.
- 3 Carefully evaluate whether you need this command for your upgrade.
- 4 To activate the command, click in the check box in the Inactive column.

The following occurs:

- The check mark disappears from the check box, indicating the command is active.
- The word Changed appears next to the check box to indicate the change was made in this session.
- The time displayed in the Net Cost column remains N/A.
- When you save and exit, Upgrade Tuner activates the command in the SQL file.
- The next time you start Upgrade Tuner, N/A is replaced by the running time for the command, and the word Changed does not appear.

- 5 Write down the SQL file name and SQL tag number for the command.
- 6 The next time you run Upgrade Tuner, locate the command and write down its net cost.

You can use a spreadsheet to keep track of net cost changes, if you prefer.

## Evaluating Upgrade Performance Improvement

To evaluate production upgrep performance improvement, add together the net cost of all the zero-row SQLs you deactivated. Then subtract the net cost of the zero-row SQLs you activated.

The final sum is an estimate of how much you have reduced the time required for the next production upgrep.

### Related Topics

[“About Tuning Siebel Production Upgrade Files” on page 109](#)

# Transferring UNIX Files for Use by Siebel Upgrade Tuner

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** UNIX only.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Upgrade Tuner is part of the Siebel Server and runs only under Windows. To tune production upgrade files on a UNIX machine you must do the following:

- Transfer upgrade files needed by Upgrade Tuner from the UNIX host to a Windows host on which a Siebel Server is installed.
- Run Upgrade Tuner on the Windows host using the UNIX files as input.
- Transfer the modified upgrade files from the Windows host back to the UNIX host.

Scripts generated by the Logparse utility during the production upprep on the UNIX host simplify the file transfer process:

- **upgtuner\_ftp\_get.txt.** This script moves upgrade files from a UNIX host to a target directory on the Windows host.
- **upgtuner\_ftp\_put.txt.** This script moves the upgrade files from the Windows host to a target directory on the UNIX host.

### Prerequisites for the UNIX Host

- In the production test environment, you must have completed the production upprep step.
- You must have run the logparse utility on the production upprep log files. See [“Summarizing Siebel Log Files Using the Logparse Utility” on page 219.](#)

### Prerequisites for the Windows Host

- You must have installed a Siebel Server. You do not have to install the Siebel Database Server.
- The Windows host can be a Siebel Server on which have performed upgrades. Tuning upgrade files from a UNIX host does not interfere with upgrade files already on the Windows-based Siebel Server.
- To run the upgtuner\_ftp\_put.txt script, you must be able to FTP from the Windows host to the UNIX host.

The following procedures use FTP to transfer files. If FTP is not available, you can use other methods for transferring files.

## Transferring Files from the UNIX Host to the Windows Host

To run Upgrade Tuner on UNIX upgrade files, you must first transfer the files to a Windows host.

### To transfer files from the UNIX host to the Windows host

**1 Windows host.** Create a target directory for the UNIX upgrade files, and share the directory.

**2 UNIX host.** FTP the following scripts to the Windows machine target directory:

- upgtuner\_ftp\_get.txt
- upgtuner\_ftp\_put.txt

The files are located in \$SIEBEL\_ROOT/bin.

**3 Windows host.** In both scripts, replace placeholder parameters with actual values, as described in the following table.

Placeholder	Value
&HostIP	This value is the IP address of UNIX machine.
&Username	This value is the username used to open an FTP session with the UNIX machine (for example sadmin).
&WindowsTempDir	This value is the full path of the target directory on the Windows machine. The target directory does not have to be within the Siebel Server installation. Avoid using a target directory that already contains upgrade files.

**4 Windows host.** Use FTP and upgtuner\_ftp\_get.txt to move the files shown in the following table from the UNIX host to the target directory on the Windows host.

File	Location on the UNIX Host
summary.xml	\$SIEBEL_ROOT/Log/upgrep_prod_ <i>VERSION</i> For example, \$SIEBEL_ROOT/Log/upgrep_prod_77/summary.xml
master_upgrep_prod_ <i>VERSION</i> .ucf	\$SIEBEL_ROOT/bin/ For example, \$SIEBEL_ROOT/bin/master_upgrep_prod_77.ucf
schema*.ddl	<i>DBSRVR_ROOT/DBPLATFORM</i> For example, <i>DBSRVR_ROOT/Oracle/schema.ddl</i> , <i>schema_t1.ddl</i> , <i>schema_t2.ddl</i>

File	Location on the UNIX Host
dri ver_upgrep_prod_ <i>VERSION</i> .ucf	<i>DBSRVR_ROOT/DBPLATFORM/upgrade/VERSION/</i> For example, <i>DBSRVR_ROOT/Oracl e/upgrade/V7_7/</i> dri ver_upgrep_prod_77.ucf
*.sql	<i>DBSRVR_ROOT/DBPLATFORM/upgrade/VERSION/</i> For example, <i>DBSRVR_ROOT/Oracl e/upgrade/V7_7/pret.sql</i> , <i>preschm.sql</i>

- 5 **Windows host.** Navigate to the target directory containing the UNIX upgrade files, and open the `summary.xml` file in a text editor.
- 6 **Windows host.** Near the beginning of the file, locate the element `<SIEBEL_ROOT>`, and edit the value to be the absolute path to the target directory containing the UNIX files that you copied to the Windows host.
- 7 **Windows host.** Save the file, and exit.
- 8 **Windows host.** Start Upgrade Tuner, and tune the UNIX upgrade files.

Specify the target directory containing the UNIX upgrade files. The `summary.xml` file contains a flag that tells Upgrade Tuner to look for all the upgrade files in the target directory. You do not have to move the files.

## Transferring Files from the Windows Host to the UNIX Host

After you have tuned the UNIX upgrade files, transfer them back to the UNIX host.

### *To transfer files from the Windows host to the UNIX host*

- 1 **UNIX host.** Create a target directory for the UNIX upgrade files that will be transferred from the Windows host.

Alternative: Use the FTP upload directory for the UNIX host.

- 2 **Windows host.** FTP the UNIX upgrade files from the target directory to the UNIX host.
- 3 **UNIX host.** Move the upgrade files to their proper locations.

The path for `<SIEBEL_ROOT>` in the `summary.xml` file is used for the Windows host and thus is incorrect for the UNIX host. The next time you run `logparse`, it will overwrite `summary.xml` and include the path for `SIEBEL_ROOT` on the UNIX host.

### Related Topics

[“About Tuning Siebel Production Upgrade Files” on page 109](#)

# Rolling Back Siebel Upgrade Tuner Changes

**Upgrades from:** All Siebel releases.

**Environments:** Production test environment only. Does not apply to production environment.

**Platforms:** Windows and UNIX only.

**Databases:** All databases.

This topic is part of an upgrade process. See [Chapter 3, “How to Perform a Siebel Database Upgrade.”](#)

Use these procedures to discard the changes you saved from the most recent Upgrade Tuner session. You do this by rolling back the upgrade files to a previous Upgrade Tuner session.

This roll-back process is particularly useful for UNIX users. You can roll back upgrade files to a previous version on the UNIX host. You do not have to transfer the files to a Windows host and rerun Upgrade Tuner.

## Upgrade File Versions

Before Upgrade Tuner saves changes to the upgrade files, it does the following:

- If this your first Upgrade Tuner session, Upgrade Tuner saves the current driver and SQL files to `.orig`, for example `driver_upgrep_prod_77.ucf.orig`.
- If this is the second session or all later sessions, Upgrade Tuner saves the current driver and SQL files to `.old`, for example `driver_upgrep_prod_77.ucf.old`.

To roll back, you replace the upgrade file with the `.old` or `.orig` version.

## Roll-Back Guidelines

Use the following guidelines to roll back upgrade files:

- To roll back to the previous Upgrade Tuner session, replace the driver or SQL file with the `.old` version.
- To roll back to the original version of the file, before any Upgrade Tuner modifications, replace the driver or SQL file with the `.orig` version.
- You can roll back the driver and SQL files separately. For example, you can roll-back to the original driver file while retaining the most recent changes to SQL files.
- You do not have to roll back all SQL files at once. For example, you can roll back some SQL files while retaining the most current version of others.
- You do not have to roll back all the commands in an SQL file. You can edit the file and activate or deactivate as many commands as desired.
- Manually editing the `driver_upgrep_prod_VERSION.ucf` file is not recommended.

- If you roll back to a session with fewer threads, you do not have to delete any `schema_t#.ddl` or `schema_i#.ddl` thread-files. The deletion is unnecessary because Upgrade Tuner removes from the driver file the steps that execute the deleted threads.

The following procedures use `dri ver_upgprep_prod_77.ucf` and `pret.sql` as examples.

### Rolling Back to the Previous Session

Use this procedure to discard your most-recent session and roll back to the previous session.

#### *To roll back to the previous session*

- 1 Save a copy of `dri ver_upgprep_prod_77` and `pret.sql` to new names.
- 2 Copy `dri ver_upgprep_prod_77.old` to `dri ver_upgprep_prod_77`.
- 3 Copy `pret.sql.old` to `pret.sql`.
- 4 Restart Upgrade Tuner.

### Rolling Back to the Original Upgrade Files

Use this procedure to discard all Upgrade Tuner changes and roll back to the original upgrade files.

#### *To roll back to the original upgrade files*

- 1 Save a copy of `dri ver_upgprep_prod_77` and `pret.sql`.
- 2 Copy `dri ver_upgprep_prod_77.orig` to `dri ver_upgprep_prod_77`.
- 3 Copy `pret.sql.org` to `pret.sql`.

### Activating or Deactivating SQL Commands Manually

Use this procedure to activate or deactivate individual zero-row SQL commands by editing the SQL file.

#### *To activate or deactivate a command by editing an SQL file*

- 1 Save a copy of the SQL file to a new name.
- 2 Open the `.sql` file (not the copy) and locate the desired SQL command.  
Commands begin `Run_SQL_#`, for example `Run_SQL_100`.
- 3 Edit the command as follows:
  - To activate the command, delete the element `(Execute=N)`
  - To deactivate a command, add the element `(Execute=N)`  
Insert the element on a line by itself after `"Run_SQL_# ="`.
- 4 Save the file.

**Related Topics**

[“About Tuning Siebel Production Upgrade Files” on page 109](#)

# A

## Siebel Marketing Upgrade Reference

This appendix contains the following topics:

- ["Obsolete Siebel Marketing Data" on page 357](#)
- ["Obsolete Business Objects in Siebel Marketing" on page 363](#)
- ["Obsolete or Replaced Views in Siebel Marketing" on page 364](#)
- ["Hidden Fields in Siebel Marketing" on page 373](#)

### Related Topics

[Chapter 3, "How to Perform a Siebel Database Upgrade"](#)

## Obsolete Siebel Marketing Data

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

[Table 67 on page 357](#) lists marketing data that is obsolete as of Siebel 7.7. In the table, the Comments field explains how the data is handled.

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Programs (program occurrences)	Obsolete	None	None	Activities migrated to parent program.
Stages (stage occurrences)	Obsolete			

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Campaigns (campaign occurrences)	Obsolete	None	DBM Campaign Campaign Occurrences Campaign Occurrences Delete Campaign Results Campaign Segment Allocation Campaign Template Related Event Templates Campaign Wave List Distribution DBM Campaign Cost DBM Campaign Cost (Fixed) DBM Campaign Cost (Inbound) DBM Campaign Cost (Outbound) DBM Campaign Occurrence Cost DBM Campaign Occurrence Lists Campaign Occurrence Offer DBM Preview List	A campaign load record is created for each occurrence.

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
eNewsletter Offers	Obsolete		Enewsletter Offer Enewsletter Offer Attachments Enewsletter Offer Attachments.Sequence Number (Sequence) Enewsletter Offer Comm Profile Parameter Enewsletter Offer Related URLs Enewsletter Offer Template Related Web Offers f/ Enewsletter Offer Related Web Surveys f/ Enewsletter Offer	Newsletter functionality is supported by conditional content formatting in email offers.
Segments	Usage history in programs is preserved.  Segment criteria are obsolete.	S_CALL_LST_CRIT S_CALL_LST_DTL S_CALL_LST_ORY	Segment Detail-DD Segment Expression Campaign Segment Allocation Segment Campaign Allocation	Segments should be reconstructed in the Segment Designer.
Filters	Obsolete	S_DD_FILTER S_DD_FILTER_DTL	Filters Filters Detail Filters Expression Save Attribute Level Look Up	Filters should be applied using criteria in the Segment Designer or Segment Tree Designer or metadata constraints in the Siebel Analytics repository.

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
List Formats	Obsolete	S_DD_LST_FMT S_DD_LST_FMTDTL S_DD_PROGLSTFMT	List Columns List Columns.Sequence (Sequence) List Formats Program List Formats Dependent List Format	Create List Formats using the List Format Designer.
Source Code Formats	Obsolete	None	None	Old source code formats are obsolete. After upgrade, source code formats need to be re-created with new elements.
Tables Joins	Obsolete	S_DD_DATA_OBJ S_DD_UNION_MBR S_DD_JOIN S_DD_JOIN_SPEC	DD Field DD Join DD Table Fields One Level Join Fields Join Fields. Sequence (Sequence) Union Tables	All data source metadata is managed in the Siebel Analytics repository.

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
List Measures Bound Measures Custom Measures	Obsolete	S_DD_MEAS S_DD_MEASRSTRCT S_DD_SUBMEAS	Aggregation Function Available Measures Base Measure Bound Measures Custom Measures Dependent List Format Functions (Logical) Functions (Mathematical) List Measures Measures Parent Measures RowNum Measure SubMeasures Measure Aggregation (Tree) Members.Sequence (Sequence) Measure Restrict Measure Restrict (Tree)	Replaced by formulas in the Siebel Analytics repository.
Buckets (Measure Base Attributes)	Obsolete	S_DD_MEASATRP S_DD_MEAS_ATTR	Measure Attribute Members Measure Attribute Members (Tree) Measure Attribute Measure Attributes Measure Attributes (Tree)	Replaced by formulas in the Siebel Analytics repository.

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Attributes and Attribute Families	Obsolete	S_DD_ATTRFAMVMB	Attribute Families	Segment criteria are now created using presentation columns ("fields") in the Segment Designer.
		S_DD_ATTRFAM	Attribute Family Resync VBC	
		S_DD_ATTRFAMLVL	Attribute Family Value	
		S_DD_ATTRFAMVAL	Attribute Filters Detail	
		S_DD_FIELD	Attribute Levels	
		S_DD_HIERATR	Attribute Levels.Sequence (Sequence)	
		S_DD_HIERATRKEY	Hierarchical Attribute Value	
		S_DD_HIERATRLVL	Hierarchical Attributes	
		S_DD_HIERATRVAL	Hierarchy (Fields)	
			Hierarchy (Tree)	
			Hierarchy Levels (Tree)	
			Hierarchy Levels Search (Tree)	
			Hierarchy Search (Fields)	
			Hierarchy Search (Tree)	

Table 67. Obsolete Siebel Marketing Data as of Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Customer Hierarchies	Obsolete	S_DD_STRFAM S_DD_SRTFAM_LVL S_DD_SRTFAM_MAP	Hierarchy Hierarchy Levels Hierarchy Levels (Fields) User Defined Levels User Defined Levels.Sequence (Sequence)	No longer necessary. Customer-level targeting is supported by Target Levels in the Siebel Analytics repository.
Campaign Load Mappings (Contact Key Formats)	Obsolete			
Server tables (internal use)	Obsolete	S_DD_CUBE S_DD_DIM S_DD_DIM_DTL S_DD_EXTRACT S_DD_EXTRACT_CL S_DD_PROGEXPORT S_DD_SNPSHT_FMT S_DD_PRPERFMEAS	Snapshot Elements	These tables were used by obsolete Siebel Server Components (Data Dictionary Manager and Marketing Server).

## Obsolete Business Objects in Siebel Marketing

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x. **Platforms:** MS Windows, UNIX, IBM z/OS.

[Table 68 on page 363](#) lists the Siebel Repository business objects that are obsolete as of Siebel 7.7.

Table 68. Obsolete Siebel Marketing Business Objects as of Siebel 7.7

Business Object Name	Tools Project
Customer Hierarchies	Server (DD)
DBM Campaign	eMarketing - Campaign Views
DBM Campaign Btn	eMarketing - Campaign Views

Table 68. Obsolete Siebel Marketing Business Objects as of Siebel 7.7

Business Object Name	Tools Project
DBM Financial Modeler	Mktg Financial Rollup
Data Dictionary	Data Dictionary (DBM)
External Contacts Mapping?	Server (DD)
Filters	Filters (DBM)
Hierarchical Attributes	Attributes (DBM)
Measure Attributes	Attributes (DBM)
Measures	Measures (DBM)
Segment	Segment (DBM)

## Obsolete or Replaced Views in Siebel Marketing

**Upgrades from:** Siebel 7.0.x & 7.5.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

[Table 69 on page 364](#) lists the Siebel 7.x Siebel Repository views that are obsolete or have been replaced as of Siebel 7.7.

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
All Campaigns (DBM) View	Campaign Management Screen	All Campaign Plans across Organizations	Replaced	Campaign Administration List
All Decisions List View	Decision Administration	All Decisions	Obsolete	
All Measures View	Marketing Administration	All Measures	Obsolete	
All Segments across My Organizations	Segment Screen	All Segments	Replaced	All Marketing Segments View
All Tables View	Marketing Administration	Tables	Obsolete	
Attribute Families View	Marketing Administration	Attribute Families	Obsolete	

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Bound Measures View	Marketing Administration	Bound Measures	Obsolete	
Campaign (DBM) Activities View	Campaign Occurrence Screen	Activities	Replaced	Campaign Activity View Admin
Campaign (DBM) Offers View	Campaign Management Screen	Offers	Replaced	Campaign Literature View Admin
Campaign (DBM) Team View	Campaign Management Screen	Team	Inactivated	
Campaign (DBM) Template Activities View	Campaign Management Screen	Activity Plans	Replaced	Campaign Activity Plans View
Campaign Admin Lists (SCW)	Campaign Screen (SCW)	Lists	Replaced	Campaign Admin Lists
Campaign Admin Objectives (SCW)	Campaign Screen (SCW)	Quotas	Replaced	Campaign Admin Objectives
Campaign Administration Detail (SCW)	Campaign Screen (SCW)	Contacts/Prospects	Replaced	Campaign Contacts Admin View - Org
Campaign Administration Detail - Owner Audit Trail View (SCW)	Campaign Screen (SCW)	Owner Audit Trail	Replaced	Campaign Administration Detail - Owner Audit Trail View
Campaign Administration Explorer View (SCW)	Campaign Screen (SCW)	Campaign Explorer	Replaced	Campaign Administration Explorer View
Campaign Administration List (SCW)	Campaign Screen (SCW)	Campaigns	Replaced	My Campaigns View
Campaign Administration List-More Info (SCW)	Campaign Screen (SCW)	More Info	Replaced	Campaign Administration List-More Info
Campaign Forecast (DBM) View-Cost Allocation Inputs	Campaign Management Screen	Cost Allocation Inputs	Replaced	Campaign Expenses View
Campaign Generated Lists View	Campaign Occurrence Screen	Exported Lists	Replaced	Campaign List Distribution View

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Campaign Groups View - Admin (SCW)	Campaign Screen (SCW)	Groups	Inactivated	
Campaign List	Campaign Screen (SCW)	Campaign List	Replaced	My Campaigns View
Campaign Literature View Admin (SCW)	Campaign Screen (SCW)	Offers	Replaced	Campaign Literature View Admin
Campaign Occurrence Contacts/ Prospects	Campaign Occurrence Screen	Campaigns > All Contacts/ Prospects across Organizations	Replaced	Campaign Contacts Admin View - All
Campaign Occurrence Contacts/ Prospects - Organization	Campaign Occurrence Screen	Campaigns > All Contacts/ Prospects	Replaced	Campaign Contacts Admin View - Org
Campaign Occurrence Contacts/ Prospects - Position	Campaign Occurrence Screen	Campaigns > My Contacts/ Prospects	Obsolete	
Campaign Occurrence Contacts/ Prospects Owner Audit View	Campaign Occurrence Screen	Campaigns > Owner Audit Trail	Replaced	Campaign Administration Detail - Owner Audit Trail View
Campaign Occurrence Email Status	Campaign Occurrence Screen	Campaigns > Email Status	Replaced	Campaign System Task View
Campaign Occurrence Offers View	Campaign Occurrence Screen	Offers	Replaced	Campaign Execution History View
Campaign Occurrence Timeline Gantt Chart View	Campaign Management Screen	Timeline	Replaced	Marketing Calendar - Campaigns Ax Gantt Chart View - My
Campaign Occurrences Responses View	Campaign Occurrence Screen	Campaigns > Responses	Replaced	Campaign Responses View
Campaign Occurrences View	Campaign Management Screen	Status	Replaced	Campaign Execution History View

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Campaign Plan Activities View	Campaign Management Screen	Activities	Replaced	Campaign Activity View Admin
Campaign Responses View (SCW)	Campaign Screen (SCW)	Responses	Replaced	Campaign Responses View
Campaign Results (DBM) Input Summary View	Campaign Management Screen	Results > Results Summary	Obsolete	
Campaign Results (DBM) View- Cost Inputs	Campaign Management Screen	Results > Cost Results	Replaced	Campaign Expenses View
Campaign Results (DBM) View-Cost Allocation Results	Campaign Management Screen	Results > Cost Allocation Results	Replaced	Campaign Expenses View
Campaign Results (DBM) View-List Results	Campaign Management Screen	Results > List Results	Replaced	Campaign Segment/ List Assumptions View
Campaign Results (DBM) View-Revenue Results	Campaign Management Screen	Results > Revenue Results	Obsolete	
Campaign Results (DBM) View-Segment Results	Campaign Management Screen	Results > Segment Inputs	Replaced	Campaign Segment/ List Assumptions View
Campaign Skill View (SCW)	Campaign Screen (SCW)	Assignment Skills	Replaced	Campaign Skill View
Campaign Team View - Admin (SCW)	Campaign Screen (SCW)	Team	Inactivated	
Campaign Waves View	Campaign Management Screen	Waves	Replaced	Program Schedule Detail View
Columns View	Marketing Administration	Fields	Obsolete	
Custom Measure Aggregation View	Marketing Administration	Custom Measure > Aggregation	Obsolete	
Custom Measure Details View	Marketing Administration	Custom Measure > More Info	Obsolete	

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Custom Measure Restriction View	Marketing Administration	Custom Measure > Restriction	Obsolete	
Customer Hierarchies View	Marketing Administration	Customer Hierarchies	Obsolete	
DBM Campaign Timeline Gantt Chart View	Campaign Management Screen	Timeline	Replaced	Marketing Calendar - Campaigns Ax Gantt Chart View - My
DD All Segments View	Segment Screen	All Segments across Organizations	Replaced	Marketing Segments Across All Organizations View
DD Segment Detail View	Segment Screen	Edit Segment	Replaced	SSO Mktg Segments Entry View
Database Synchronization	Marketing Administration	Data Retrieval	Obsolete	
Decision Wizard Player View	Decisions Administration Screen		Obsolete	
Decisions Detail Input Parameters View	Decisions Administration Screen	Identifiers	Obsolete	
Decisions Detail Output Parameters View	Decisions Administration Screen	Outputs	Obsolete	
Decisions Detail Session Parameters View	Decisions Administration Screen	Real Time Inputs	Obsolete	
Decisions Detail View	Decisions Administration Screen	More Info	Obsolete	
Enewsletter Attachments View	Offer Screen	eNewsletter > Attachments	Obsolete	
Enewsletter Email Profile View	Offer Screen	eNewsletter > Profile	Obsolete	
Enewsletter Offer Detail View	Offer Screen	eNewsletter > More Info	Obsolete	
Enewsletter Offer Related URLs View	Offer Screen	eNewsletter > Related URLs	Obsolete	

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Enewsletter Offer Sections List View	Offer Screen	eNewsletter > Sections	Obsolete	
Enewsletter Offer View	Offer Screen	eNewsletter Offers	Obsolete	
Enewsletter Rules View	Offer Screen	eNewsletter Rules	Obsolete	
External Contacts Mapping	Marketing Administration	Campaign Load Mapping	Obsolete	
Filters Detail View	Marketing Administration	Filters	Obsolete	
Filters View	Marketing Administration	Filters	Obsolete	
Financial Modeler (DBM) View	Campaign Management Screen	Forecast > Input Summary	Obsolete	
Financial Modeler (DBM) View-Fixed Costs	None	Financial Modeler	Obsolete	
Financial Modeler (DBM) View-Inbound Costs	None	Financial Modeler	Obsolete	
Financial Modeler (DBM) View-Input Costs	Campaign Management Screen	Forecast > Cost Inputs	Replaced	Campaign Expenses View
Financial Modeler (DBM) View-List Inputs	Campaign Management Screen	Forecast > List Inputs	Replaced	Campaign Segment/List Assumptions View
Financial Modeler (DBM) View-Outbound Costs	None	Financial Modeler	Obsolete	
Financial Modeler (DBM) View-Revenue Inputs	None	Financial Modeler	Obsolete	
Financial Modeler (DBM) View-Segment Inputs	Campaign Management Screen	Forecast > Segment Inputs	Replaced	Campaign Segment/List Assumptions View
Hierarchical Attribute List View	Marketing Administration	Hierarchical Attributes	Obsolete	

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Joins View	Marketing Administration	Joins	Obsolete	
List Formats View	Marketing Administration	Output List Layouts	Obsolete	
List Measures View	Marketing Administration	List Measures	Obsolete	
Marketing Plans Funds	Marketing Plans	Funds	Inactivated	
Measure Based Attributes View	Marketing Administration	Buckets	Obsolete	
My Campaigns (DBM) View	Campaign Management Screen	My Campaign Plans	Replaced	My Campaigns View
My Campaigns (DBM) View - More Info	Campaign Management Screen	My Campaign Plans	Replaced	Campaign Administration List-More Info
My Decisions List View	Decision Administration	My Decisions	Obsolete	
My Segments View	Segments	My Segments	Replaced	My Marketing Segments View
My Team's Decisions List View	Decision Administration	My Team's Decisions	Obsolete	
Program (DBM) Activities View	Program Occurrence Screen	Activities	Replaced	Program Plan Activities View
Program Container Timeline Gantt Chart View	Program Screen	Timeline	Replaced	Program Schedule Timeline View
Program Lists View	Program Stages Screen	Preview List	Replaced	Campaign List Distribution View
Program Occurrence Timeline Gantt Chart View	Program Screen	Timeline	Replaced	Marketing Calendar - Programs Ax Gantt Chart View - My
Program Occurrences View	Program Screen	Status	Obsolete	
Program Responses View	Program Occurrence Screen	Responses	Obsolete	

Table 69. Siebel 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Siebel 7.x Screen	Siebel 7.x View Caption	Outcome	Replacement View
Program Snapshots View	Program Stages Screen	Snapshot Elements	Obsolete	
Program Stages View	Program Screen	Stages	Inactivated	
Related Events View for eNewsletter Offers	Offer Screen	Related Events	Obsolete	
Related Web Offers View f/ Enewsletter	Offer Screen	Related Web Offers	Obsolete	
Related Web Surveys View f/ Enewsletter	Offer Screen	Related Web Surveys	Obsolete	
Response Detail View (SME) - More Info	Response Screen	More Info	Replaced	Response Detail View (Detail)
Segment Criteria View	Segment Screen	More Info	Obsolete	
Segment Program Stages View	Segment Screen	Program Plans	Obsolete	
eChannel All Programs View (DBM)	eChannel Program Screen - OLD	All Program Plans across Organizations	Replaced	All Programs View (DBM)
eChannel All Programs across My Organizations	eChannel Program Screen	All Programs	Replaced	All Programs across My Organizations
eChannel Campaign Administration Detail	eChannel Campaign Management Screen - OLD	Contacts/ Prospects	Replaced	Campaign Contacts Admin View - Org
eChannel My Programs View (DBM)	eChannel Program Screen - OLD	My Program Plans	Replaced	My Programs View (DBM)
eChannel Program Container Timeline Gantt Chart View	eChannel Program Screen - OLD	Timeline	Replaced	Marketing Calendar - Programs Ax Gantt Chart View - My
eChannel Programs Detail View (DBM)	eChannel Program Screen - OLD	More Info	Replaced	Programs Detail View (DBM)
eChannel Segment Program Stages View	eChannel Segment Screen	Program Plans	Obsolete	

## Obsolete Views in Siebel Marketing

**Platforms:** MS Windows, UNIX, IBM z/OS.

Table 70 on page 372 lists the Siebel 6.x Siebel Repository views that are obsolete as of Siebel 7.7.

Table 70. Siebel 6.x: Obsolete Siebel Marketing Views

Obsolete View Name	Siebel 6.x Screen	Siebel 6.x View Caption	Outcome	Replacement View
Program Lists View	Programs	Preview Lists	Obsolete	
Program Workflow Processes View	Programs	Processes	Inactivated	
Program (DBM) Activities View	Programs	Occurrences > Activities	replaced	Program Plan Activities View
Program Responses View	Programs	Occurrences > Responses	Obsolete	
Program Occurrences View	Programs	Occurrences > Status	Obsolete	
Campaign Waves View	Campaigns	Waves	Replaced	Program Schedule Detail View, Campaign List Distribution View
Campaign Generated Lists View	Campaigns	Exported Lists	Replaced	Campaign List Distribution View
Campaign Occurrences View	Campaigns	Occurrences	Replaced	Campaign Execution History View
Enewsletter Offer Detail View	Offers	eNewsletter > Detail	Obsolete	
Enewsletter Email Profile View	Offers	eNewsletter > Email Profile	Obsolete	
Enewsletter Attachments View	Offers	eNewsletter > Attachments	Obsolete	
Enewsletter Rules View	Offers	eNewsletter > Rules	Obsolete	

# Hidden Fields in Siebel Marketing

**Upgrades from:** Siebel 6.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

In Siebel Marketing, some fields that were exposed in Siebel 6.x are hidden as of Siebel 7.7. If your business processes depends on these fields, you can re-expose them. [Table 71 on page 373](#) lists the hidden fields. Obsolete fields or fields from obsolete business components are not listed.

Table 71. Siebel 6.x: Hidden Siebel Marketing Views

Buscomp Field Name	UI Caption	Business Component
Division	Division	Program Container
Region	Region (LOV)	Program Container
Parent Program Id	Parent Program	Program Container
Revenue Goal	Revenue Goal	Program Container
Revenue Prediction	Revenue Prediction	Program Container
Cost Prediction	Cost Prediction	Program Container
Revenue Goal	Revenue Goal	Program (DBM)
Budget	Budget	Program (DBM)
Revenue Prediction	Revenue Prediction	Program (DBM)
Cost Prediction	Cost Prediction	Program (DBM)
Division	Division	Campaign
Region	Region (LOV)	Campaign
Purpose	Purpose	Campaign
Target	Target	Campaign
Target Calls Per Day	Target Calls per Day	Campaign
Target Total Calls	Total Target Calls	Campaign
Calculated Sub Cost	Calculated Sub Cost	Campaign
Adjusted Cost	Adjusted cost	Campaign
Total Cost	Total Cost	Campaign



# B

## Tables Modified or Seeded During a Siebel Upgrade

This appendix includes the following topics:

- [“Tables Seeded as of Siebel 7.7” on page 375](#)
- [“Tables Modified During Upgrade from Siebel Release 7.0.x” on page 383](#)
- [“Tables Modified During Upgrade from Siebel Siebel 6.x” on page 391](#)

### Related Topics

[Chapter 3, “How to Perform a Siebel Database Upgrade”](#)

## Tables Seeded as of Siebel 7.7

**Upgrades from:** Siebel 6.x, 7.0.x, & 7.5.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

[Table 72](#) lists Siebel tables that are seeded as of Siebel 7.7.

Table 72. seeded as of Siebel 7.7

Table
S_ACTION_ARG
S_ACTION_DEFN
S_APP_QUERY
S_APP_VIEW
S_APP_VIEW_RESP
S_ASGN_GRP
S_ASGN_GRP_EMP
S_ASGN_GRP_OBJ
S_ASGN_GRP_POSTN
S_ASGN_RULE
S_ASGN_RULE_ITEM
S_ASGN_WL_OBJ
S_ASGN_WL_OBJ_COL
S_AUDIT_BUSCOMP

Table 72. seeded as of Siebel 7.7

Table
S_AUDIT_FIELD
S_BC_ENTRY_TRGT
S_BRWSR
S_BRWSR_CPBLTY
S_BRWSR_ELM_EVT
S_BRWSR_FEATURE
S_BU
S_BUSCOMP_DMAP
S_BUSOBJ_DMAP
S_CB_ASSET
S_CB_ASSET_AGRP
S_CB_ASSET_DATA
S_CB_ASSET_VER
S_CB_CNTNT_TYPE
S_CB_CNNTYP_EXT
S_CM_CFG_PROF
S_CM_CMD
S_CM_CMD_DATA
S_CM_CMD_PARM
S_CM_CMD_REL
S_CM_CNCTR
S_CM_CNCTR_PARM
S_CM_DATA_PARM
S_CM_EVTHDLR
S_CM_EVTLOG
S_CM_EVTRESP
S_CM_HDLR_PARM
S_CM_LOG_PARM
S_CM_MEDIA
S_CM_MEDIA_PROP
S_CM_PROF

Table 72. seeded as of Siebel 7.7

Table
S_CM_PROF_PARM
S_CM_RESP_LOG
S_CM_RESP_PARM
S_COLLAB_CMD
S_CONTACT
S_CONTACT_BU
S_CONTXT_BC
S_CONTXT_BC_VAR
S_CO_BUS_AREA
S_CS_ANSWR
S_CS_EDGE
S_CS_PAGE
S_CS_PATH
S_CS_PATH_BU
S_CS_PATH_SCPT
S_CS_PATH_SRC
S_CS_QUEST
S_CS_QUEST_SCPT
S_CS_STYLE
S_CTI_CFG
S_CTI_CFG_PARAM
S_CTI_EXTENSION
S_CTI_USER
S_CTLG
S_CTLG_BU
S_CTLG_CAT
S_CTLG_CAT_PROD
S_CT_ACTION
S_CT_ACTION_SET
S_CT_APLT_RLST
S_CT_APPLET

Table 72. seeded as of Siebel 7.7

Table
S_CT_EVENT
S_CT_EVENT_DEF
S_CT_RULE
S_CT_RULE_SET
S_CURCY
S_DB_SCH_VER
S_DCP_PKG_ITEM
S_DCP_RESP_TYPE
S_DOCK_ROUTE
S_DOCK_ROUTE_VR
S_DOC_CORR
S_DOC_PPSL
S_DOC_PPSL_BU
S_DUAL
S_DYNUI_APPLET
S_DYNUI_VIEW
S_EAI_ADAPTER
S_EAI_ADPT_FLOW
S_EAI_APPLVER
S_EAI_APPL_ADPT
S_EAI_DTEOBJ2MP
S_EAI_FLOW
S_EAI_LOOKUPMAP
S_EAI_QUEUE
S_EMP_PER
S_ESCL_ACTION
S_ESCL_COND
S_ESCL_GROUP
S_ESCL_RULE
S_ETL_CTRYREGN
S_ETL_DATASRC

Table 72. seeded as of Siebel 7.7

Table
S_ETL_DAY
S_ETL_ERRHLP
S_ETL_INDUS_VER
S_ETL_SRC_TABLE
S_ETL_STRINGS
S_ETL_TIME_DAY
S_EVT_ACT
S_EXCH_RATE
S_EXP_ITEM_TYPE
S_FCST_SCENARIO
S_FIELD_DMAP
S_INDUST
S_INT_CHANNEL
S_INT_COMPMAP
S_INT_FLDMAP
S_INT_GROUP
S_INT_INSTANCE
S_INT_MAP_ARG
S_INT_OBJMAP
S_INVLOC
S_INVLOC_BU
S_INVLOC_TYPE
S_INV_TXN_TYPE
S_LANG
S_LIT
S_LIT_BU
S_LOCALE
S_LOV_REL
S_LST_OF_VAL
S_MPP_FLD_MAP
S_MPP_MAP

Table 72. seeded as of Siebel 7.7

Table
S_MPP_OBJ_MAP
S_MSG
S_NEWS_PKG
S_NEWS_TOPIC
S_NODE
S_OLAP_CAT
S_OLAP_REPORT
S_OLAP_RPT_CAT
S_ONL_CATEGORY
S_ORDER_TYPE
S_ORDTYP_MVTTYP
S_ORG_EXT
S_PAGER_COMPANY
S_PARTMVMT_TYPE
S_PARTY
S_PARTY_GROUP
S_PARTY_GRP_BU
S_PARTY_PER
S_PARTY_RPT_REL
S_PERF_RTNG_SCL
S_PERIOD
S_PERIOD_BU
S_PERIOD_REL
S_PER_RESP
S_POSTN
S_POSTN_CON
S_PPSL_SECT
S_PPSL_SECT_FLD
S_PPSL_SECT_LIT
S_PROC_DEFN
S_PROD_INT

Table 72. seeded as of Siebel 7.7

Table
S_PROD_INT_BU
S_PROD_INV_CAT
S_PRTL_ITEM
S_PRTL_PAGE
S_PRTL_SCTN_ITM
S_PRTL_SECTION
S_RESP
S_RESP_BU
S_RULE_DATATYPE
S_RULE_DESIGNER
S_RULE_TMPL
S_RULE_TMPL_ARG
S_RULE_TMPL_CAT
S_SALES_METHOD
S_SALES_MTHD_BU
S_SA_APPL
S_SA_APPL_CNTNT
S_SA_APPL_FLD
S_SA_CNTNT_TYPE
S_SA_CTLG_NMVAL
S_SA_NMVAL_APPL
S_SA_NMVAL_DEF
S_SGML_DOC_TYPE
S_SM_STATE
S_SM_TRANSITION
S_SRCH_ADM
S_SRCH_FLD
S_STATE_MODEL
S_STG
S_SVCDSP_RL_PRP
S_SVCDSP_RL_SET

Table 72. seeded as of Siebel 7.7

Table
S_SVCDSP_RULE
S_SVC_METRIC
S_SYS_PREF
S_TIMEZONE
S_TMPL_PLANITEM
S_UI_CTRL_STYLE
S_UPG_COMP
S_UPG_COMP_REL
S_UPG_KIT_ITARG
S_UQ_CFG
S_UQ_CFG_PARAM
S_UQ_ROUTE
S_UQ_ROUTE_ESCL
S_UQ_ROUTE_PROP
S_UQ_RTESCL_RUL
S_USER
S_USERLIST
S_USERLIST_ATT
S_USERLIST_BU
S_USER_PROF
S_USER_ROLE_TSK
S_USER_TASK
S_VARSUB_CONTXT
S_WEB_APPL
S_WEB_ELMNT_EVT
S_WEB_ML_ELMNT
S_WEB_UI_EVENT
S_WF_COND_CRIT
S_WF_COND_VAL
S_WF_PROC_FLOW
S_WF_PROC_PROP

Table 72. seeded as of Siebel 7.7

Table
S_WF_STEP
S_WF_STEP_ARG
S_WF_STEP_BRNCH
S_WI_CTNT_FXUP
S_WI_HTTP_HOST
S_WI_SYMURL
S_WI_SYMURL_ARG
S_WS_BNDNG_DTL
S_WS_OPERATION
S_WS_PORT
S_WS_PORT_TYPE
S_WS_WEBSERVICE
S_XML_NAMESPACE
S_ZD_ENBLD_OBJ
S_ZIPCODE

## Tables Modified During Upgrade from Siebel Release 7.0.x

**Upgrades from:** Siebel 7.0.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Databases:** All databases.

Table 73 lists tables that are modified during upgrades from Release 7.0.x. Modifications are indicated in the table as follows:

- **I** = inserted records
- **U** = updated records

■ **R** = referenced records

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_ACCNT_CHRCTR	R	R
S_ACCNT_POSTN	U	U
S_ACT_EMP	UR	UR
S_ACT_ORDER	R	R
S_ACTIVITY_ATT	R	R
S_ADDR_ORG	R	R
S_AGREE_ITEM	U	U
S_AGREE_TERMS	U	U
S_APP_VER	U	U
S_APP_VIEW	R	R
S_APP_VIEW_RESP	U	U
S_ASSESS	R	R
S_ASSESS_VAL	U	U
S_ASSET	UR	UR
S_ASSET_ACCNT	I	I
S_BC_ENTRY_TRGT	UR	UR
S_BCSTMSG_DIVN	R	R
S_BRDCST_MSG	U	U
S_BU	R	R
S_CALL_LST	R	R
S_CALL_LST_BU	I	I
S_CAMP_CON	UR	UR
S_CB_ASSET	IR	IR
S_CB_ASSET_AGRP	I	I
S_CB_ASSET_DATA	I	I
S_CB_ASSET_REF	I	I
S_CB_ASSET_VER	IR	IR
S_CB_CNTNT_SET	IR	IR
S_CB_CNTTYP_EXT	R	R

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_CHRCTR	U	U
S_COLUMN	UR	UR
S_COMMUNICATION	U	U
S_CON_CHRCTR	R	R
S_CONTACT	UR	UR
S_CONTACT_BU	I	I
S_CONTACT_INFO	R	R
S_CONTACT_REL	U	U
S_CONTACT_T	U	U
S_CRSE_JOBRL	UR	UR
S_CRSE_OFFR_TST	UR	UR
S_CRSE_TST	U	U
S_CRSE_TST_LANG	R	R
S_CRSE_TST_QUES	R	R
S_CRSE_TSTRUN	R	R
S_CRSE_TSTRUN_Q	U	U
S_CS_PATH	U	U
S_CS_QUEST	U	U
S_CURRCLM	R	R
S_CURRCLM_CRSE	U	U
S_CURRCLM_JOBRL	UR	UR
S_CURRCLM_STEP	I	I
S_CUST_SURVEY	U	U
S_DD_HIERATR	U	U
S_DD_HIERATRVAL	U	U
S_DEAL_DCP	R	R
S_DMND_CRTN_PRG	U	U
S_DNB_ORG	U	U
S_DOC_AGREE	R	R
S_DOC_ORDER	U	U
S_DOC_QUOTE	UR	UR

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_DOCK_TXN_LOG	R	R
S_DOCK_VIS_RULE	U	U
S_DUAL	R	R
S_EIM_FK_MAP	UR	UR
S_EIM_FK_MAPCOL	UR	UR
S_EMP_INDUST	R	R
S_EMP_PER	U	U
S_EMP_PERF_MEAS	U	U
S_ESCL_REQ	R	R
S_EVT_ACT	UR	UR
S_EVT_FUL_REQ	U	U
S_EVTLOC	IUR	IUR
S_EVTLOC_REL	I	I
S_EXP_ITEM	U	U
S_EXP_ITEM_TYPE	R	R
S_EXP_RPT	U	U
S_FCST	R	R
S_FCST_ITEM	U	U
S_FCSTSER_DATE	R	R
S_FUL_REQ_CNTNT	R	R
S_IC_CALC	U	U
S_IC_CALC_IT	U	U
S_IC_CALC_RUN	U	U
S_IC_CMPNT_FLTR	I	I
S_IC_DRAW	R	R
S_IC_DRAW_BAL	I	I
S_IC_DRAW_PER	U	U
S_IC_PAYMENT	UR	UR
S_IC_PERF_HIST	U	U
S_IC_PLCMP_PER	IR	IR
S_IC_PLN_CMPNT	R	R

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_IC_RATE	IUR	IUR
S_IC_TXN	UR	UR
S_IC_TXN_IT	U	U
S_INDUST	R	R
S_INT_COMP	R	R
S_INT_COMPMAP	R	R
S_INT_FLDMAP	R	R
S_INT_OBJ	R	R
S_INT_OBJMAP	R	R
S_INVOICE_ITEM	U	U
S_JOB_PROFILE	IR	IR
S_LIST_COLUMN	U	U
S_LIT	R	R
S_LIT_PRTL_DISP	R	R
S_LOCALE	R	R
S_LST_OF_VAL	UR	UR
S_NEWS_TOPIC	I	I
S_NEWSTPC_USER	I	I
S_ONL_OBJECT	U	U
S_ONL_PS_CTGRY	R	R
S_OPTY	U	U
S_OPTY_CHRCTR	R	R
S_OPTY_CON	U	U
S_OPTY_ORG	R	R
S_ORD_CRDT_ASGN	I	I
S_ORDER	UR	UR
S_ORDER_DTL	R	R
S_ORDER_ITEM	U	U
S_ORDER_POSTN	R	R
S_ORDER_SHIP	R	R
S_ORG_BU	U	U

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_ORG_EXT	UR	UR
S_ORG_EXT_T	U	U
S_ORG_GROUP	UR	UR
S_ORG_GROUP_BU	I	I
S_ORG_PRTNR	U	U
S_ORG_PRTNR_ATT	R	R
S_ORG_PRTNR_RGN	R	R
S_ORG_REL	R	R
S_ORGGRP_POSTN	U	U
S_PARTY	UR	UR
S_PARTY_GROUP	I	I
S_PARTY_GRP_BU	I	I
S_PARTY_PER	R	R
S_PER_COMM_ADDR	R	R
S_PER_WRK_EXP	R	R
S_PERF_RVW	R	R
S_PERF_RVW_CMP	R	R
S_PERF_RVW_COMP	I	I
S_PERF_RVW_TMPL	IUR	IUR
S_PERIOD	R	R
S_POOL_QUES	I	I
S_POSTN	UR	UR
S_PPSL_SECT	U	U
S_PRD_DFCT_EMP	R	R
S_PRD_DFCT_TAG	R	R
S_PRI_LST_ITEM	U	U
S_PROD_CFGVER	R	R
S_PROD_DEFECT	U	U
S_PROD_INT	IUR	IUR
S_PROD_INT_CRSE	I	I
S_PROJ	IUR	IUR

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_PROJ_BU	I	I
S_PROJ_RSRC	IR	IR
S_PROJITEM	IUR	IUR
S_PROJITEM_ATT	I	I
S_PROJITEM_ORG	I	I
S_PROJITEM_REL	R	R
S_PROJITEM_RSRC	I	I
S_PROJITEM_TAG	R	R
S_PROJRSRC_SKL	U	U
S_PROJRSRC_SKLI	U	U
S_PRSP_CONTACT	U	U
S_PRTL_DISP	R	R
S_PRTL_ITEM	IUR	IUR
S_PRTL_ITM_AGRP	I	I
S_PRTL_PAGE	IUR	IUR
S_PRTL_PG_AGRP	I	I
S_PRTL_SCTN_ITM	IR	IR
S_PRTL_SECTION	IUR	IUR
S_QTA_INCENTV	UR	UR
S_QTA_OBJ	R	R
S_QTA_PLAN	R	R
S_QTA_POSTN	UR	UR
S_QTAOBJAS_PER	U	U
S_QTASGN_POSTN	U	U
S_QUOTE_ITEM	U	U
S_REPOSITORY	UR	UR
S_RESP	R	R
S_REVN	R	R
S_RVW_COMP	IU	IU
S_SCHM_PHASE	R	R
S_SCHM_PHS_USG	R	R

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_SCHM_PROC	R	R
S_SCHM_PROC_VAR	R	R
S_SCHM_STEP	R	R
S_SCHM_STEP_DEP	R	R
S_SCHMPROC_DBMS	R	R
S_SCHMST_DBSCPT	R	R
S_SCHMSTEP_SCPT	R	R
S_SCREEN	R	R
S_SHIPMENT	U	U
S_SRC	IUR	IUR
S_SRC_BU	I	I
S_SRC_COST	U	U
S_SRC_DCP	R	R
S_SRC_EVT	IUR	IUR
S_SRC_EVT_REG	U	U
S_SRC_ORG	R	R
S_SRC_PAYMENT	IUR	IUR
S_SRC_VENUE	IR	IR
S_SRCH_CTRESFLD	U	U
S_SRCH_INDX_FLD	U	U
S_SRM_ACT_PARAM	R	R
S_SRM_ACTION	R	R
S_SRV_ACT	I	I
S_SRV_REQ	R	R
S_SRV_REQ_BU	I	I
S_SYS_PREF	U	U
S_TABLE	UR	UR
S_TMPL_PLANITEM	R	R
S_TMSHT	U	U
S_TMSHT_ITEM	U	U
S_TST_QUES	R	R

Table 73. Tables Modified During Upgrades from Release 7.0.x

Table	7.0.3	7.0.4
S_UK_ATTJOIN	R	R
S_USER	IR	IR
S_USER_KEY	UR	UR
S_USER_KEY_ATT	UR	UR
S_USERLIST	I	I
S_USERLIST_BU	I	I
S_USR_MSG	U	U
S_WEB_CNTNT	R	R
S_WEB_CNTNT_LNK	R	R
S_WF_PROC_FLOW	U	U
S_WF_STEP	U	U
S_XA_CLASS	U	U

## Tables Modified During Upgrade from Siebel Siebel 6.x

**Upgrades from:** Siebel 6.x.

**Platforms:** MS Windows, UNIX, IBM z/OS.

**Databases:** All databases.

Table 74 lists tables that are modified during upgrades from Siebel 6.x. Modifications are indicated in the table as follows:

- **I** = inserted records
- **U** = updated records
- **R** = referenced records

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_ACCELERATOR	IR	IR	IR	IR
S_ACCELRTTR_INTL	I	I	I	I
S_ACCNT_ATT	I	I	I	I
S_ACCNT_CHRCTR	R	R	R	R

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_ACCNT_POSTN	IUR	IUR	IUR	IUR
S_ACT_CONTACT	IR	IR	IR	IR
S_ACT_EMP	IUR	IUR	IUR	IUR
S_ACT_ORDER	R	R	R	R
S_ACT_TIMESTAMP	R	R	R	R
S_ACTIVITY_ATT	R	R	R	R
S_ADDR_ORG	IUR	IR	IR	IR
S_ADDR_ORG_INT	UR	R	R	R
S_ADDR_PER	IUR	IR	IR	IR
S_AGREE_ITEM	U	U	U	U
S_AGREE_POSTN	IR	IR	IR	IR
S_APP_VER	U	U	U	U
S_APP_VIEW	R	R	R	R
S_APP_VIEW_RESP	U	U	U	U
S_APPL_WTMPL_IT	U	U	U	U
S_APPLET	R	R	R	R
S_ASGN_ATTR_COL	R	R	R	R
S_ASGN_RULE	U	U	U	U
S_ASSESS	R	R	R	R
S_ASSESS_VAL	U	U	U	U
S_ASSET	UR	UR	UR	UR
S_ASSET_ACCNT	IR	IR	IR	IR
S_ASSET_WRNTY	R	R	R	R
S_AUC_ALRT_CAT	R	R	R	R
S_AUC_ALRT_DEF	U	U	U	U
S_AUC_ITEM	U	U	U	U
S_BC_ENTRY_TRGT	UR	UR	UR	UR
S_BC_VIEWMODE	I	I	I	I
S_BCSTMSG_DIVN	R	R	R	R
S_BRDCST_MSG	U	U	U	U
S_BU	IUR	IUR	IUR	IUR

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_BUSCOMP	UR	UR	UR	UR
S_C_LST_CHRCTR	I	I	I	I
S_C_LST_COL_MAP	R	R	R	R
S_C_LST_MKT_SEG	R	R	R	R
S_C_LST_POSTN	R	R		
S_C_LST_QRY_CAT			R	R
S_CALL_LST	UR	UR	UR	UR
S_CALL_LST_BU	I	I	I	I
S_CALL_LST_TMPL	I	I	I	I
S_CAMP_CON	UR	UR	UR	UR
S_CERT_ATTR_PER	R	R	R	R
S_CFG_CLASS	IR	IR	IR	IR
S_CFG_ITEM	R	R	R	R
S_CFG_PROP_DEF	IUR	IUR	IUR	IUR
S_CFG_RULE_DEF	I	I	I	I
S_CFG_RULE_SPEC	R	R	R	R
S_CFG_SCRIPT	R	R	R	R
S_CFG_SCRPT_DEF	IR	IR	IR	IR
S_CFG_VAR_DEF	IUR	IUR	IUR	IUR
S_CFGA_VARDEF	R	R	R	R
S_CHRCTR	IU	IU	IU	IU
S_CHRCTR_VAL	I	I	I	I
S_CM_CFG_USER	IR	IR	IR	IR
S_CM_TELESET	IR	IR	IR	IR
S_CM_USER_QUE	IR	IR	IR	IR
S_CO_BUS_AREA	U	U	U	U
S_COLUMN	UR	UR	UR	UR
S_COMM_REQ			U	U
S_COMM_TMPL	R	R	R	R
S_COMMAND	R	R	R	R
S_COMMUNICATION	U	U	U	U

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_CON_ADDR	R	R	R	R
S_CON_CHRCTR	IR	IR	IR	IR
S_CON_MKT_SEG	R	R	R	R
S_CONTACT	IUR	IUR	IUR	IUR
S_CONTACT_ATT	I	I	I	I
S_CONTACT_BU	I	I	I	I
S_CONTACT_INFO	R	R	R	R
S_CONTACT_REL	U	U	U	U
S_CONTACT_SS	U	U	U	U
S_CONTACT_T	U	U	U	U
S_CONTACT_X	U	U	U	U
S_CONTROL	U	U	U	U
S_CRSE	R	R	R	R
S_CRSE_AUD	UR	UR	UR	UR
S_CRSE_LIT	UR	UR	UR	UR
S_CRSE_OFFR	UR	UR	UR	UR
S_CRSE_OFFR_TST	IUR	IUR	IUR	IUR
S_CRSE_REG	R	R	R	R
S_CRSE_REL	U	U	U	U
S_CRSE_TST	UR	UR	UR	UR
S_CRSE_TST_LANG	R	R	R	R
S_CRSE_TST_QUES	UR	UR	UR	UR
S_CRSE_TSTRUN	U	U	U	U
S_CS_PATH	UR	UR	UR	UR
S_CS_QUEST	U	U	U	U
S_CS_RUN	U	U	U	U
S_CTI_CFG	R	R	R	R
S_CTI_EXTENSION	R	R	R	R
S_CTI_TELESET	R	R	R	R
S_CTI_USER	R	R	R	R
S_CTLG	UR	UR	UR	UR

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_CTLG_CAT	UR	UR	UR	UR
S_CTLG_CAT_AUC	R	R	R	R
S_CTLG_CAT_CRSE	R	R	R	R
S_CTLG_CAT_ISS	R	R	R	R
S_CTLG_CAT_LIT	R	R	R	R
S_CTLG_CAT_ORG	R	R	R	R
S_CTLG_CAT_PROD	UR	UR	UR	UR
S_CTLG_CAT_SR	R	R	R	R
S_CTLG_CAT_SRC	R	R	R	R
S_CTLGCT_RESITM	R	R	R	R
S_CURRCLM	R	R	R	R
S_CURRCLM_CRSE	U	U	U	U
S_CURRCLM_PER	U	U	U	U
S_CURRCLM_STEP	I	I	I	I
S_CUST_SURVEY	U	U	U	U
S_DD_HIERATR			U	U
S_DD_HIERATRVAL			U	U
S_DD_MEAS_ATTR			IR	IR
S_DD_PROGLSTFMT	R	R		
S_DEAL_DCP	R	R	R	R
S_DMND_CRTN_PRG	IUR	IUR	IUR	IUR
S_DNB_CON_MRC	R	R	R	R
S_DNB_CONTACT	R	R	R	R
S_DNB_ORG	U	U	U	U
S_DOC_AGREE	UR	UR	UR	UR
S_DOC_AGREE_BU	I	I	I	I
S_DOC_ORDER	U	U	U	U
S_DOC_PPSL	R	R	R	R
S_DOC_PPSL_BU	I	I	I	I
S_DOC_QUOTE	UR	UR	UR	UR
S_DOC_QUOTE_BU	I	I	I	I

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_DOCK_TXN_LOG	R	R	R	R
S_DOCK_VIS_RULE	U	U	U	U
S_DUAL	R	R	R	R
S_EIM_FK_MAP	UR	UR	UR	UR
S_EIM_FK_MAPCOL	UR	UR	UR	UR
S_EMP_INDUST	R	R	R	R
S_EMP_PER	IUR	IUR	IUR	IUR
S_EMP_PERF_MEAS	U	U	U	U
S_EMPLOYEE_ATT	R	R	R	R
S_EMPLOYEE_X	R	R	R	R
S_EMPLOYEE_XM	R	R	R	R
S_ESCL_REQ	UR	R	R	R
S_ESCL_RULE	R			
S_EVT_ACT	UR	UR	UR	UR
S_EVT_ACT_X	R	R	R	R
S_EVT_FUL_REQ	U	U	U	U
S_EVT_MAIL	I	I	I	I
S_EVT_STAFF	R	R	R	R
S_EXP_ITEM	U	U	U	U
S_EXP_ITEM_TYPE	R	R	R	R
S_EXP_RPT	U	U	U	U
S_FIELD	R	R	R	R
S_FUL_REQ_CNTNT	R	R	R	R
S_IC_CALC	UR	U	U	U
S_IC_CALC_IT	UR	U	U	U
S_IC_CALC_RUN	IUR	U	U	U
S_IC_CMPNT_FLTR	I	I	I	I
S_IC_DRAW		R	R	R
S_IC_DRAW_BAL	I	I	I	I
S_IC_DRAW_DIVN		I	I	I
S_IC_DRAW_PER		IU	IU	IU

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_IC_FLTR	U			
S_IC_PAYMENT	UR	UR	UR	UR
S_IC_PERF_HIST	U	U	U	U
S_IC_PLCMP_DIVN		I	I	I
S_IC_PLCMP_PER	IR	IR	IR	IR
S_IC_PLN_CMPNT	UR	UR	UR	UR
S_IC_RATE	IUR	IUR	IUR	IUR
S_IC_TXN	UR	UR	UR	UR
S_IC_TXN_IT	UR	U	U	U
S_IC_TXN_POSTN	U			
S_INDUST	R	R	R	R
S_INT_CKEY	IUR	IUR	IUR	IUR
S_INT_CKEY_FLD	I	I	I	I
S_INT_COMP	UR	UR	UR	UR
S_INT_COMPMAP	R	R	R	R
S_INT_FIELD	UR	UR	UR	UR
S_INT_FLDMAP	R	R	R	R
S_INT_OBJ	R	R	R	R
S_INT_OBJMAP	R	R	R	R
S_INV_TXN	R	R	R	R
S_INV_TXN_BU	IR	IR	IR	IR
S_INVLOC	UR	UR	UR	UR
S_INVOICE_ITEM	U	U	U	U
S_ISS	R	R	R	R
S_ISS_BU	I	I	I	I
S_LIST_COLUMN	U	U	U	U
S_LIT	UR	UR	UR	UR
S_LIT_PRTL_DISP			I	I
S_LOCALE	R	R	R	R
S_LST_OF_VAL	UR	UR	UR	UR
S_MBL_DVC	I	I	I	I

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_MBL_DVC_ADDR	I	I	I	I
S_MDF	R	R	R	R
S_MDF_ALLOC	R	R	R	R
S_MDF_BU	IR	IR	IR	IR
S_MKT_SEG	UR	UR	UR	UR
S_MKT_SEG_VAL	R	R	R	R
S_NEWS_TOPIC	I	I	I	I
S_NEWSTPC_USER	I	I	I	I
S_NOTE	R	R	R	R
S_NOTE_DCP	IR	IR	IR	IR
S_NOTE_USR_MSG	I	I	I	I
S_OLAP_CAT	U	U	IUR	IUR
S_OLAP_REPORT			IR	IR
S_OLAP_RPT_CAT			I	I
S_OLAPCAT_POSTN	R	R	IR	IR
S_ONL_OBJ_PARTY	R	R	R	R
S_ONL_OBJECT	U	U	U	U
S_ONL_PROJ_PRTY	R	R	R	R
S_ONL_PROJECT	U	U	U	U
S_ONL_PS_CTGRY	R	R	R	R
S_OPTY	UR	UR	UR	UR
S_OPTY_BU	I	I	I	I
S_OPTY_CHRCTR	R	R	R	R
S_OPTY_INDUST	R	R	R	R
S_OPTY_ORG	R	R	R	R
S_OPTY_POSTN	UR	UR	UR	UR
S_OPTY_PROD	R	R	R	R
S_OPTY_PROJ	R	R	R	R
S_ORD_CRDT_ASGN	I	I	I	I
S_ORDER	UR	UR	UR	UR
S_ORDER_BU	IU	IU	IU	IU

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_ORDER_DTL	R	R	R	R
S_ORDER_ITEM	UR	UR	UR	UR
S_ORDER_POSTN	IUR	IUR	IUR	IUR
S_ORDER_SHIP	R	R	R	R
S_ORDER_TYPE	R	R	R	R
S_ORG_BU	I	I	I	I
S_ORG_CRDT_AREA	R	R	R	R
S_ORG_EXT	IUR	IUR	IUR	IUR
S_ORG_EXT_SS	U	U	U	U
S_ORG_EXT_T	UR	UR	UR	UR
S_ORG_EXT_X	U	U	U	U
S_ORG_FUL	U	U		
S_ORG_FUL_PER	R	R		
S_ORG_GROUP	UR	UR	UR	UR
S_ORG_GROUP_BU	I	I	I	I
S_ORG_GROUP_X	U	U	U	U
S_ORG_INT_ATT	R	R	R	R
S_ORG_MKT_SEG	R	R	R	R
S_ORG_PRTNR	U	U	U	U
S_ORG_PRTNR_ATT	R	R	R	R
S_ORG_PRTNR_EXP	R	R	R	R
S_ORG_PRTNR_RGN	R	R	R	R
S_ORG_REL	R	R	R	R
S_ORGGRP_POSTN	UR	UR	UR	UR
S_ORGGRP_TERR	R	R	R	R
S_OU_PRTNR_TIER	IR	IR	IR	IR
S_OU_PRTNR_TYPE	I	I	I	I
S_PARTY	IUR	IUR	IUR	IUR
S_PARTY_PER	IR	IR	IR	IR
S_PARTY_REL	U	U	U	U
S_PARTY_RPT_REL	IR	IR	IR	IR

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_PER_COMM_ADDR	R	R	R	R
S_PER_PRTNRAPPL	U	U	U	U
S_PER_RESP	IR	IR	IR	IR
S_PER_WRK_EXP	R	R	R	R
S_PERIOD	R	R	R	R
S_PICKLIST	U	U	U	U
S_POSTN	UR	UR	UR	UR
S_POSTN_CON	I	I	I	I
S_PPSL_SECT	U	U	U	U
S_PRD_DFCT_EMP	R	R	R	R
S_PRD_DFCT_TAG	R	R	R	R
S_PRDFCT_PRJITM	IR	IR	IR	IR
S_PRDFCT_SYMP	R	R	R	R
S_PREPAY_BAL	U	U	U	U
S_PREPAYBAL_PTY	R	R	R	R
S_PRI_LST	UR	UR	UR	UR
S_PRI_LST_BU	IR	IR	IR	IR
S_PRI_LST_ITEM	U	U	U	U
S_PRILST_MKTSEG	R	R	R	R
S_PRIMDL_FCTR	U	U	U	U
S_PROD_CFGVER	IR	IR	IR	IR
S_PROD_DEFECT	UR	UR	UR	UR
S_PROD_DFCT_BU	I	I	I	I
S_PROD_INT	IUR	IUR	IUR	IUR
S_PROD_INT_BU	U	U	U	U
S_PROD_INT_CRSE	I	I	I	I
S_PROD_ITEM	IUR	IUR	IUR	IUR
S_PROJ	IUR	IUR	IUR	IUR
S_PROJ_BU	I	I	I	I
S_PROJ_POSTN	I	I	I	I
S_PROJ_RSRC	IR	IR	IR	IR

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_PROJITEM	IUR	IUR	IUR	IUR
S_PROJITEM_ATT	I	I	I	I
S_PROJITEM_DIVN	R	R	R	R
S_PROJITEM_ORG	I	I	I	I
S_PROJITEM_REL	R	R	R	R
S_PROJITEM_RSRC	IR	IR	IR	IR
S_PROJITEM_TAG	R	R	R	R
S_PRSP_CONTACT	IUR	IUR	IUR	IUR
S_PRTL_DISP			IR	IR
S_PRTNRAPP_REGN	R	R	R	R
S_PRTNRAPPL_EXP	R	R	R	R
S_PRTNRAPPL_IND	R	R	R	R
S_PRTNRAPPL_MKT	R	R	R	R
S_PTSHP_MKT_SEG	R	R	R	R
S_QRY_CAT			R	R
S_QTA_ACHIEVE	U			
S_QTA_INCENTV	UR	UR	UR	UR
S_QTA_OBJ	UR	UR	UR	UR
S_QTA_OBJASGN	R	R	R	R
S_QTA_OBJCRT	U	U	U	U
S_QTA_ORG_INT		R	R	R
S_QTA_PLAN	UR	R	R	R
S_QTA_POSTN	UR	UR	UR	UR
S_QTAOBJAS_DIVN		I	I	I
S_QTAOBJAS_PER	U	IU	IU	IU
S_QTASGN_POSTN	U	U	U	U
S_QUOTA	R	R	R	R
S_QUOTE_ITEM	U	U	U	U
S_QUOTE_SOLN	R	R	R	R
S_REPOSITORY	UR	UR	UR	UR
S_RESITEM	U	U	U	U

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_RESP	UR	UR	UR	UR
S_REVN	IR	IR	IR	IR
S_RULE_DESIGNER	R	R	R	R
S_RULE_TMPL	U	U	U	U
S_SALES_METHOD	R	R	R	R
S_SALES_MTHD_BU	IR	IR	IR	IR
S_SHIPMENT	U	U	U	U
S_SRC	IUR	IUR	IUR	IUR
S_SRC_BU	I	I	I	I
S_SRC_COST	U	U	U	U
S_SRC_DCP	IR	IR	IR	IR
S_SRC_EVT	IUR	IUR	IUR	IUR
S_SRC_EVT_REG	IUR	IUR	IUR	IUR
S_SRC_INDUST	R	R	R	R
S_SRC_PAYMENT	IUR	IUR	IUR	IUR
S_SRC_POSTN	I	I	I	I
S_SRC_VENUE	R	R	R	R
S_SRM_ACT_PARAM	R	R	R	R
S_SRM_ACTION	R	R	R	R
S_SRM_REQUEST	R	R	R	R
S_SRV_ACT	I	I	I	I
S_SRV_REQ	UR	UR	UR	UR
S_SRV_REQ_BU	I	I	I	I
S_STG_LIT	R	R	R	R
S_STORE_COND	U	U	U	U
S_SYMPTOM	R	R	R	R
S_SYS_PREF	U	U	U	U
S_TABLE	UR	UR	UR	UR
S_TERR	R			
S_TMPL_PLANITEM	R	R	R	R
S_TMPL_PLNIT_BU	IR	IR	IR	IR

Table 74. Tables Modified During Upgrades from Siebel 6.x

Table	6.0.1	6.2.1	6.3	6.3.1
S_TMSHT	U	U	U	U
S_TMSHT_ITEM	UI	UI	UI	UI
S_TST_QUES	I	I	I	I
S_UK_ATTJOIN	R	R	R	R
S_USER	IR	IR	IR	IR
S_USER_KEY	UR	UR	UR	UR
S_USER_KEY_ATT	UR	UR	UR	UR
S_USR_MSG	IR	IR	IR	IR
S_USR_MSG_ATT	I	I	I	I
S_USR_MSG_CAT	I	I	I	I
S_USR_MSG_PREF	I	I	I	I
S_VIEW	U	U	U	U
S_VIEW_WTMPL_IT	U	U	U	U
S_WEB_PG_ITM	U	U	U	U
S_WEB_TMPL	R	R	R	R
S_WEB_TMPL_FILE	I	I	I	I
S_WF_PROC_FLOW	U	U	U	U
S_WF_STEP	U	U	U	U
S_WF_STEP_ARG	U	U	U	U
S_WRNTY_CVRG	R	R	R	R
S_WRNTY_CVRG_BU	IR	IR	IR	IR
S_XA_ATTR	IR	IR	IR	IR
S_XA_CLASS	IUR	IUR	IUR	IUR
S_XA_CLASS_REL	I	I	I	I



# C

## Upgrading to eConfigurator Version 7

The appendix contains the following topics:

- [“What’s New in eConfigurator Version 7” on page 405](#)
- [“Overview of Upgrading to eConfigurator Version 7” on page 411](#)
- [“Upgrading from eConfigurator Version 6.x to eConfigurator Version 7” on page 412](#)
- [“Approach 1: Using the Standard Upgrade Tool” on page 412](#)
- [“Approach 2: Modifying Your Existing Implementation” on page 438](#)
- [“Upgrading from Release 5.x Product Configurator to eConfigurator Version 7” on page 440](#)

### What’s New in eConfigurator Version 7

This chapter provides the information necessary for upgrading to server-based eConfigurator version 7, from either the Release 6.x Product Configurator or the Release 5.x Product Configurator.

To help you better understand the upgrade issues in deciding which upgrade approach to use, [“What’s New in eConfigurator Version 7” on page 405](#) highlights some of the major new features in eConfigurator version 7.

Siebel 7 eConfigurator provides many new features that you will want to take into consideration when planning your upgrade process. [Table 75](#) lists the key eConfigurator features that have been enhanced, added, or replaced in version 7. These new features are highlighted to assist you in your upgrade planning.

Table 75. Siebel eConfigurator Version 7 Features

Feature	Where to Find Information
<p><b>Product Attributes and Classification System</b></p> <p>In earlier versions of Siebel, eConfigurator rules could be used in association with virtual products and properties to create the notion of product attributes. Now, with the new attributes and product classification system, the need for virtual products, properties, and certain associated configuration rules can in these instances be met more efficiently. Attributes now can be set and managed as part of product definition and can be set at run time. In these cases, you should consider restructuring your data and use the new attributes and product classification system.</p>	<p>For more information, see <a href="#">“Product Attributes and Classification System”</a> on page 409 and <a href="#">“Customizable Products”</a> on page 409. Also see <i>Product Administration Guide</i>.</p> <p>For information on restructuring data, see <a href="#">“Review and Adjust eConfigurator Version 6.x Models (Recommended)”</a> on page 416.</p>
<p><b>Customizable Products</b></p> <p>Like product attributes, in earlier versions of Siebel a combination of Siebel eConfigurator rules, virtual products, and categories were often used to create product relationships. With eConfigurator version 7, customizable products and the Product Rule Manager allow the creation of bills of material (BOMs), hierarchical products, as well as nested- and subconfigurations. Data that was structured in eConfigurator version 6.x to mimic this type of functionality will not translate exactly into the eConfigurator version 7 framework. Also, since the customizable-product structure can present products and their associated prices in a hierarchical format, the customer's ability to review a quote is enhanced, eliminating the need for the quote solution provided in eConfigurator version 6.x.</p>	<p>For more information, see <a href="#">“Customizable Products”</a> on page 409. Also see <i>Product Administration Guide</i>.</p> <p>For information on restructuring product configurations, see <a href="#">“Review and Adjust eConfigurator Version 6.x Models (Recommended)”</a> on page 416.</p>

Table 75. Siebel eConfigurator Version 7 Features

Feature	Where to Find Information
<p><b>Natural Language Rule Design</b></p> <p>Siebel eConfigurator version 7's natural-language Product Rule Manager allows nontechnical users to administer simple and complicated configuration rules.</p>	<p>For more information, see <a href="#">"Product Rule Manager" on page 410</a>.</p> <p>For information on reviewing rules of an upgraded configuration model, see <a href="#">"Checking Configuration Rules (Required)" on page 424</a>.</p> <p>Also see <i>Product Administration Guide</i>.</p>
<p><b>UI Flexibility</b></p> <p>Siebel eConfigurator version 7 allows administrators to customize the application user interface without using Siebel Tools. During the upgrade process, default UI themes will be applied to any upgraded customizable products. To fully realize the benefit of this feature, you may want to modify or redesign certain UI elements.</p>	<p>For more information, see <a href="#">"Declarative User Interface Customization" on page 410</a>. Also see <i>Product Administration Guide</i>.</p>
<p><b>Multiple Instantiation</b></p> <p>With eConfigurator version 7, you can now configure multiple instances of the same product simultaneously within a single configuration session. You may want to consider restructuring your upgraded configuration models in order to make use of this feature.</p>	<p>For more information, see <a href="#">"Multiple Instantiation" on page 410</a>. Also see <i>Product Administration Guide</i>.</p> <p>For information on restructuring configuration models, see <a href="#">"Review and Adjust eConfigurator Version 6.x Models (Recommended)" on page 416</a>.</p>
<p><b>Browser-Based Configuration</b></p> <p>Since you will receive both the server-based and browser-based deployment options of eConfigurator version 7 through the upgrade process, you may find that the browser-based deployment more appropriately matches your business needs. The browser-based deployment of eConfigurator represents an additional upgrade option available to you.</p>	<p>For more information, see <a href="#">"Browser-Based Configuration" on page 411</a>.</p> <p>Browser-based deployment is described in <i>Siebel Interactive Designer Administration Guide</i>.</p> <p>Also see <i>Product Administration Guide</i>.</p>

The development of Siebel's new features often required coming up with a new approach to solving a particular business problem. This has resulted in significant changes to the underlying structure and orientation of Siebel eConfigurator.

Because many of the new features in eConfigurator version 7 approach business problems in a manner that could not have been possible in Release 6.x, there are some limitations to the upgrade process, which are outlined in ["Limitations" on page 427](#). Like any upgrade, the eConfigurator upgrade process cannot guarantee that the new features in the application are used as effectively as possible after the upgrade has taken place, since the legacy data may not be structured in a way to make use of these features. To fully realize the value of the new features offered in eConfigurator version 7, you may want to consider restructuring select parts of your implementation.

The sections that follow provide more detail about the major enhancements in eConfigurator version 7. This information is provided to assist you in planning your upgrade. For more comprehensive information about new Siebel 7 features, or eConfigurator version 7 features, see *Product Administration Guide*.

## Product Attributes and Classification System

The product classification system new to eConfigurator version 7 simplifies product administration by shrinking the number of products that require maintenance.

You no longer have to create a single product record and configuration rule for each product combination. In eConfigurator version 7, you define products once in the product catalog, then append multiple attributes (and associated price adjustments, using Siebel ePricer). When you define product classes, you assign attributes to the classes or use the default attributes (see ["Default Values"](#) below), and then these attributes are inherited from parent class to subclasses.

## Default Values

Siebel eConfigurator version 7 introduces the concept of default values to product administration. This simplifies the assigning of attributes to your products, and it also simplifies ordering for your customers. By setting default values when defining your products, you give your customers the option of configuring their product choices, or simply choosing the defaults you have set up for them.

## Customizable Products

Using eConfigurator version 7, product administrators can create customizable products as part of the product definition process rather than having to define relationship rules. You define products and their components in multilevel hierarchies, along with related quantities. You define parent-child relationships among products, facilitating both fixed and substitutable product bundles, bills of materials, and multilevel subconfigurations, or nested configurations.

## Product Rule Manager

The improved rules designer offered by eConfigurator version 7 provides a graphic designer for building eConfigurator rules—both simple and complex. Using rule templates—stored sets of rules—you can build other rules or templates for future reuse. The rule templates are created using a natural-language expression that includes predefined placeholders for items or attributes.

**NOTE:** A new rules-design paradigm was required to create this enhanced functionality. Certain rules designed within eConfigurator version 6.x, particularly those built using the Product Rule Manager, will not match the rule structure of eConfigurator version 7. After the upgrade, you should review these rules for accuracy, and you may need to restructure them. See [“Checking Configuration Rules \(Required\)”](#) on page 424.

## Declarative User Interface Customization

Siebel eConfigurator version 7 provides flexibility to customize the application UI to meet your business needs, without using Siebel Tools. You can create your own selection pages in a configuration session for each configurable product and service by selecting from a palette of predefined controls and associating these controls to items that comprise the product definition. For example, an administrator setting up the display for a PC can choose to display CPU options using a drop-down list and show hard-drive options as a series of radio buttons, and then choose from an assortment of predefined templates to provide a logical grouping of the various options that comprise the PC. You can also define your own controls and themes. The UI customizations are not hard-coded; they are created using administrative tools in a data-driven (declarative) fashion. They can be modified using either the run-time administration screens or by changing HTML templates.

## Multiple Instantiation

Siebel eConfigurator version 7 provides the ability to manage complex configuration requirements, including the ability to configure multiple instances of the same customizable product simultaneously within a single configuration run-time session. Multiple instantiation allows for more complete modeling of the relationships between the items in a complex configuration. Each customizable product instance can have different values for its selections, such as quantity or attribute values.

## Browser-Based Configuration

Siebel eConfigurator operates both on a server-based technology (described throughout this chapter) and a browser-based technology, which performs the processing directly in the end user's Web browser. When an end user initiates a Siebel eConfigurator session, the system downloads user-interface elements, the Siebel eConfigurator browser-based engine, and specific browser-based Siebel eConfigurator data to the user's browser. Once downloaded, all processing can be done on the user's local device, thereby eliminating the need for round trips to the server after every user request. As these are downloaded, processing is done on the user's local device, eliminating the need for round trips to the server after every user selection or request. The result is an interactive session that allows the user to perform configurations and dynamically view the results of selections. This solution is particularly useful when deployed to large numbers of users without concern for bandwidth and latency.

You create new Siebel eConfigurator browser-based models using the Siebel Interactive Designer administration environment, which allows administrators to capture and describe both the data and logic of the browser-based Siebel eConfigurator deployment as well as define the run-time user interface. Once completed, the administrator can then selectively choose the Siebel eConfigurator models to be exposed to end users. For more information, see *Siebel Interactive Designer Administration Guide*.

## Overview of Upgrading to eConfigurator Version 7

In upgrading to eConfigurator version 7, you have two major options to choose from when deciding on the approach you will take. You can choose to use the Siebel upgrade tool, or you can choose to manually reimplement your existing Siebel application natively within Siebel 7.

The first approach will migrate as much configuration data as you defined and created in the previous releases to the new application environment. At the same time, it will put fewer of the new Siebel 7 features to work for you. The second approach can give you the full benefits of the new Siebel 7 features, but with limitations on reusing (reconfiguring) existing data such as quotes.

Choosing the appropriate upgrade approach requires careful examination of your business practices as well as thorough consideration of your existing data and the model data you would like to have. In some circumstances, a combination of the two approaches may provide the optimal result. If you need assistance analyzing and planning your upgrade, please contact Siebel Professional Services or the Siebel ISS Competency Center group.

This chapter covers the two upgrade approaches in [“Upgrading from eConfigurator Version 6.x to eConfigurator Version 7” on page 412](#). If you are upgrading from Release 5.x to Siebel 7 eConfigurator, you should also read [“Upgrading from Release 5.x Product Configurator to eConfigurator Version 7” on page 440](#). The upgrade from Release 5.x is very similar to that of Release 6.x, so the two separate upgrade approaches are available in this case as well. If you are upgrading from Release 5.x, you will take the same steps as those upgrading from Release 6.x.

**NOTE:** For feature mappings between eConfigurator version 6.x and eConfigurator version 7, see the *Product Administration Guide*. The chapter on mapping features between eConfigurator version 6.x and eConfigurator version 7 provides mapping information for administration features, run-time features, quote integration, scripts, and methods. That chapter lists every eConfigurator version 6.x feature and its closest corresponding Siebel 7 feature. For those features not supported in Siebel 7, you will find alternate methods that implement the same or similar functionality. Once you understand the differences and similarities of these features, you will be able to migrate or design your model data and business logic more clearly and effectively.

## Upgrading from eConfigurator Version 6.x to eConfigurator Version 7

In upgrading from eConfigurator version 6.x to eConfigurator version 7, you can choose between two major upgrade options: you can use the Siebel standard data upgrade tool and modify the upgraded configuration models, or you can redesign your models natively within the eConfigurator version 7 environment to make best use of the new features.

**NOTE:** For whichever upgrade option you choose, you must conduct the standard Siebel upgrade to migrate to the Siebel 7 application and database environment.

### Approach 1: Using the Standard Upgrade Tool

Before you use the upgrade tool, you may want to review your configuration models and verify some product settings so that the upgrade process will perform optimally. Then you will run the standard upgrade tool, and your eConfigurator model and data will upgrade automatically. There are also some recommended preupgrade and postupgrade tasks to help you complete the process more effectively.

#### Overview

Siebel System’s standard upgrade program migrates existing Release 6.x data into the Siebel 7 database and data repository. The Release 6.x data migrated includes configuration models and configuration data in quotes. As much as possible, the data upgrade script Siebel provides preserves the modeling information that you previously entered when designing your models. It does this by systematically mapping and translating similar data items.

However, since the design paradigms in eConfigurator version 6.x and eConfigurator version 7 are different, some of the functions of the version 6.x features may be implemented differently with Siebel 7 features and are therefore not directly applicable to the Siebel 7 upgraded models. See the *Product Administration Guide* for feature mapping information. Also, the customizable products generated during the standard upgrade cannot make full use of the new Siebel 7 features, such as classification systems and dynamic attributes, without substantial adjustments being made after the automatic upgrade process.

This section describes the details of what the upgrade program does. You will also find recommended procedures to follow before and after performing the upgrade.

Figure 9 shows the process flow of upgrading to eConfigurator version 7 following Approach 1, using the Siebel upgrade tool and reviewing the upgraded models. Solid boxes represent required steps and dotted boxes show recommended steps.

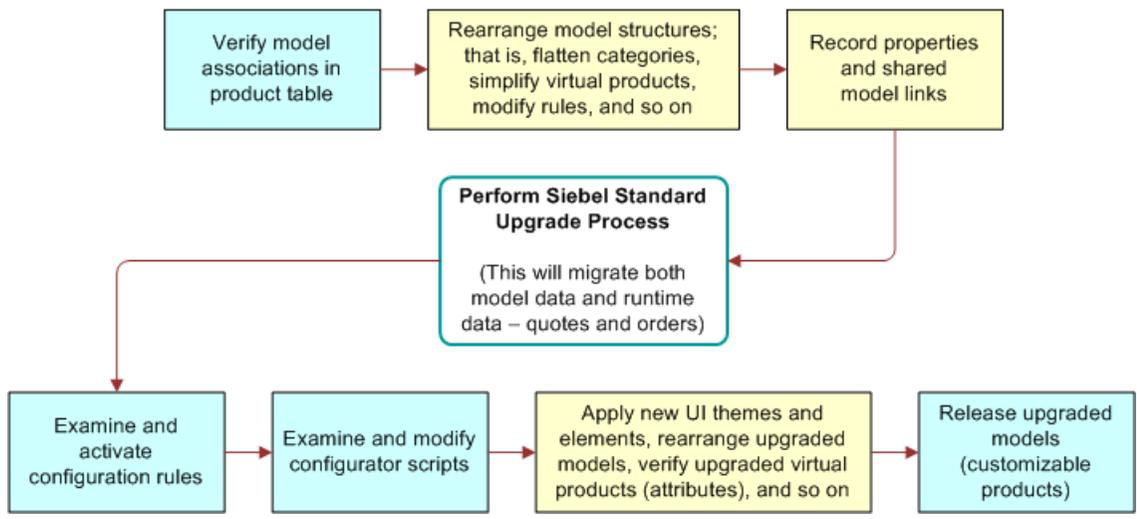


Figure 9. Overview of Upgrading to Siebel eConfigurator Version 7 (Approach 1)

## Preupgrade Tasks

This section identifies a series of tasks to complete before launching the automatic Siebel upgrade script.

**NOTE:** Since the Configurator upgrade script is part of the Siebel standard upgrade script, you need to complete the tasks in this section prior to beginning the overall Siebel application upgrade.

Using these procedures, the Siebel upgrade script can preserve most of your modeling information during the upgrade process. The script will apply some new Siebel 7 features on top of the upgraded data. After running the upgrade script, you should review the upgraded data before it is released to use. See [“Postupgrade Tasks” on page 424](#).

## Verifying the Model Associations in the Product Table (Required)

In addition to migrating standard Siebel products definitions, the Siebel upgrade script will migrate configuration model data for related model products instead of migrating the models directly. The upgrade script does this by upgrading model products that have the Model Product field checked and the Configuration Model Name field populated with associated model names.

To make sure your configuration model definitions are properly preserved, before the entire upgrade process starts you should review your product table and verify that appropriate model products are created and associated with their corresponding configuration models.

### *To verify the model associations*

- 1 Navigate to Screens > Configuration Designer > Model Manager.
- 2 Record all the models for which you want to preserve definitions.
- 3 Navigate to Screens > Marketing Administration > Products > Products view.

- 4 In the Products list applet, do the following:
  - a Search for all products with the Model Product field checked and the Configuration Model Name field filled in.
  - b Decide whether these model products will continue to be used as Siebel 7 customizable products. If so, make sure they have proper models associated with them.
  - c For model products that are not going to be used:
    - Uncheck the Model Product field.
    - Remove the association of the configuration model name.

**NOTE:** Removing the association of a configuration model name has implications on the quotes upgrade. For example, a Release 6.x model product could be used to create a quote with a solution. If the model association remains across the upgrade process, the solution in the quote will be upgraded as a customizable product quote line, which will be read-only and can be reconfigured by eConfigurator version 7. If the model product's association is removed, then the upgrade script will upgrade the quote solution into a package product, and each line item in the solution will be a component of this package product. In this case, this specific solution will remain directly editable but not reconfigurable by eConfigurator version 7. So you should only remove associations if you want to relinquish the configurability of the existing quotes that were created with a model product. This will likely be appropriate if you plan to use the second upgrade option (Approach 2) to natively create your configuration models.

- 5 Make sure every model for which you want to preserve data has at least one model product associated with it.
  - a If a model does not have a model product associated with it, stay in Marketing Administration > Products > Products to create a product and make the association.

## Review and Adjust eConfigurator Version 6.x Models (Recommended)

The Siebel upgrade script will migrate your Release 6.x configuration model products into Siebel 7 customizable products with similar structures. However, since the design paradigms for models and customizable products are different, it is recommended that you review each of your Release 6.x models, analyzing its use under the new eConfigurator features. The following steps will help you review and make necessary adjustments to your model data and to make best use of the upgrade script to migrate your data into the new Siebel 7 environment.

### *To review and adjust eConfigurator version 6.x model data*

For each Release 6.x configuration model:

- 1 Flatten the structure so that there is, at most, one level of products or categories.

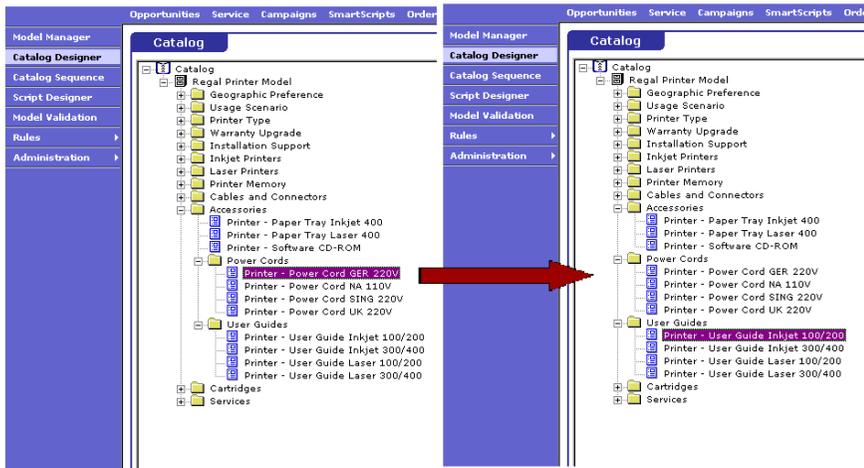
The customizable product feature, new in Siebel 7, uses nested customizable products to achieve a bill-of-material structure. Use this functionality to create multiple-level products, rather than the eConfigurator version 6.x category structure. Also, you can use the new UI functionality to create groups and tabs to achieve a flexible user interface and to help the end user navigate to find the right product. Instead of letting the upgrade script flatten the structure for you, you may rearrange your product components now in order to keep maximum control of the outcome, and in order to plan for adopting the new features.

**NOTE:** Flattening your model structure brings the following considerations:

When you change your structures, some of the rules may become invalid. You may need to redefine those rules accordingly in the post-upgrade environment.

In any way that you decide to flatten your models, you should avoid having a product appear multiple times under the same top-level category. This could cause problems in the Siebel 7 environment when reconfiguring an upgraded customizable product quote.

Be cautious when removing products from your models because the corresponding quotes will not be read properly by eConfigurator after the upgrade. This is also the case for Release 6.x as these products will be missing when revalidating existing quotes with the latest version of the model.



**2** Start planning your new class structure when reorganizing your model.

The upgrade script will not automatically create classes, as the same product may belong to different categories but not different classes. Since the concept of class is so important in the Siebel 7 customizable product structure, you should group products that may be of the same class into the categories you are going to keep (the top-level categories). This way you can assign classes to appropriate relationships after upgrading so as to make use of the new classification feature. See [“Using the New eConfigurator Version 7 Features \(Recommended\)” on page 426](#).

**3** Record all model properties.

In Release 6.x, properties were used locally in configuration models. Their main usage has been replaced by the Siebel 7 feature of dynamic attributes. The upgrade script will not migrate properties, so you may want to record these local properties and start planning their attribute replacements. For more details about how to implement dynamic attributes, see *Product Administration Guide*. Also see [“Using the New eConfigurator Version 7 Features \(Recommended\)” on page 426](#).

**NOTE:** Some configuration rules that referenced these properties will become invalid too. You should revisit them in the postupgrade process. See [“Checking Configuration Rules \(Required\)” on page 424](#).

**4** Examine the virtual products in the models.

- a If some virtual products were used to simulate product attributes, consider leaving them as they are or consolidating them.

An example of consolidating a set of virtual products might be that for a category “Color” with three virtual products—Blue, White, and Black—you remove the category “Color” and keep only one virtual product. Change the virtual product name to “Color.” Now after the upgrade, the appropriate attribute will be created and the attribute-simulating function of the three virtual products will be preserved by extending the attribute domain (postupgrade).

**NOTE:** If you consolidate virtual products, some rules that reference them will become invalid. You may have to recreate these rules in the postupgrade environment. See “Checking Configuration Rules (Required)” on page 424 and “Verifying Attributes Upgraded from Virtual Products (Recommended)” on page 425.

- b If virtual products were used for other purposes, consider removing them and using other new eConfigurator features to implement the functions.

For example, it is recommended that you use Siebel eAdvisor to conduct needs analysis previously done through the use of configuration models. See *Product Administration Guide* and *Siebel Interactive Designer Administration Guide* for more information.

## What the Upgrade Script Does

In order to make use of the new Siebel 7 features and reduce future development and maintenance difficulty, the Siebel upgrade script will inevitably make trade-offs when migrating the existing configurator data. Understanding how the script performs the upgrade and the rationale behind it should help you to better manage your new configuration information in the Siebel 7 environment.

**NOTE:** See “Limitations” on page 427 for more information about possible upgrade mismatches.

The Siebel upgrade script takes the following steps to transform your existing Release 6.x model information to Siebel 7 corresponding data:

### Release 6.x eConfigurator Model Data

For every Release 6.x model product that is associated with a configuration model, the Siebel upgrade script:

- Creates a new class with the same name. This will be a top-level class.

- Creates for every virtual product in the associated model a new attribute with the same name, assigning it to the class created above. It sets the attribute type as Integer and the default value as 0. No other attribute domain is defined.

**NOTE:** All references to “associated model” in this section refer to the last released configuration model that is associated to the model product.

- Creates a customizable product with the same name and assigned to the class created above. With this, the customizable product inherits the newly created attributes (from virtual products).

Configuration Model Name	Model Product	Product	Part Num
Nomad Performance Laptop M...	✓	<a href="#">Nomad Performance Laptop</a>	NomadPe
PCS Computer Shared Model v...	✓	<a href="#">PCS Computer System</a>	PCSCOMP
Performa Workstation Model	✓	<a href="#">Performa 800 Workstation</a>	PCS6880C
Regal Performance Workstatio	✓	<a href="#">Regal Performance Workstation</a>	RegalPerf
Regal Printer Model	✓	<a href="#">Regal Printers</a>	PCSREGAI
Regal Pro Workstation Model	✓	<a href="#">Regal Pro Workstation</a>	RegalPro

Name	Type	Display Name
Nomad Performance Laptop Model	Product	Nomad Performance Laptop Model
PCS Computer Shared Model with Links	Product	PCS Computer Shared Model with Links
Performa Workstation Model	Product	Performa Workstation Model
Regal Performance Workstation Model	Product	Regal Performance Workstation Model
Regal Printer Model	Product	Regal Printer Model
Regal Pro Workstation Model	Product	Regal Pro Workstation Model

A customizable product can preserve most of the structure and business logic of a Release 6.x model. In all the remaining bullet points below, you will find descriptions of model elements being migrated to their corresponding upgraded customizable product.

**NOTE:** Model product names are preserved during the upgrade process, but model names are not. If you rely on Release 6.x model names, you should rename their corresponding model products to preserve the proper information.

- Creates for every top-level category in the associated model a class relationship with the same name and attaches the relationship to the customizable product created. The cardinality of the relationship is set to a minimum of zero and a maximum of infinity, meaning there is no limitation of a certain quantity.

The upgrade script does not create a new class for the relationship. This is to minimize the creation of redundant and possibly confusing classes. You should carefully plan your class system and apply your classes accordingly after the upgrade process.

- Creates for every top-level product in the model a product relationship with the same name and attaches to the customizable product. The cardinality of the relationship is set to a minimum of zero and a maximum of infinity. The upgrade script does not create and assign a class to the product.

**NOTE:** The upgrade script does not create product relationships for virtual products.

- Creates for every product in a lower-level category a product relationship under its top-level class relationship (converted from a Release 6.x top-level category).

This step and the previous steps transform the model structure into the new customizable product structure. The upgrade script creates only a single-level relationship in the new customizable product. This is because multi-level customizable products in Siebel 7 are created by nesting customizable products, one within another.

**NOTE:** If you have completed Step 1 in [“Review and Adjust eConfigurator Version 6.x Models \(Recommended\)”](#) on page 416 you will have flattened your models so that they have only one level, and this step will not take place.

- For every shared model, do the following:
  - a If the shared model is at the top level, create a relationship in the root model and put all the products in the shared model into this relationship.
  - b If the shared model is in a category, all the products in the shared model are placed in the new relationship that is created for that category.

- C If there are nested shared models (Model A contains Model B, which contains model C), the upgrade script recursively adds the products from the shared models to the relationship. The script supports a maximum of 10 levels of nesting.

Relationship Display Name	Class Name	Product	Define Domain
Geographic Preference	Regal Printer Model		
Usage Scenario	Regal Printer Model		
Printer Type	Regal Printer Model		
Warranty Upgrade	Regal Printer Model		
Installation Support	Regal Printer Model		
Inkjet Printers	Regal Printer Model		
Laser Printers	Regal Printer Model		

**AND**

Relationship Display Name	Class Name	Product	Define Domain
Accessories	Regal Printer Model		
Printer - Paper Tray Inkjet 400	Regal Printer Model	Printer - Paper Tray Inkjet 400	
Printer - Paper Tray Laser 400	Regal Printer Model	Printer - Paper Tray Laser 400	
Printer - Software CD-ROM	Regal Printer Model	Printer - Software CD-ROM	
Printer - Power Cord NA 110V	Regal Printer Model	Printer - Power Cord NA 110V	
Printer - Power Cord SING 220V	Regal Printer Model	Printer - Power Cord SING 220V	
Printer - Power Cord UK 220V	Regal Printer Model	Printer - Power Cord UK 220V	

- Copies all configuration rules (basic and advanced) from the associated model as they are into the Siebel 7 customizable product. The upgrade script sets the rules to “Inactive.”

Siebel eConfigurator version 7 has a different design paradigm that may alter your strategy for rule creation. As described above, proper use of the new eConfigurator features will potentially reduce the number of rules in many instances. Also, the new eConfigurator uses “path” to identify a configuration item rather than “CfgId” (used in Release 6.x).

**NOTE:** You will have to revisit every rule to make sure it functions correctly after the upgrade. You can do this by using the Advanced Rule and Assisted Advanced Rule templates in the Product Rule Manager. For more information, see [“Checking Configuration Rules \(Required\)”](#) on page 424.

- Upgrades the scripts in the associated Release 6.x model into the Siebel 7 customizable product as they are.

**NOTE:** Since scripts are redesigned in Siebel 7, you will need to revisit these scripts and determine their proper usage after the upgrade. See [“Verifying Scripts \(Required\)”](#) on page 425. For more information, see the scripts mapping table in *Product Administration Guide*.

- Upgrades linked items in the associated Release 6.x model into the Siebel 7 customizable product as they are. In eConfigurator version 7, the linked items work the same way.

- Upgrades the resources in the associated model into the resources in the new customizable product. In eConfigurator version 7, the resources work the same way.

The new customizable product will use the Siebel 7 default UI templates (base theme, product theme, and UI controls) to display the selection pages in a configuration session.

**NOTE:** The new customizable product will not be released automatically. You should revisit the upgraded customizable product before you release it to production. See “Postupgrade Tasks” on page 424 and “Validating and Releasing Upgraded Customizable Products (Required)” on page 426.

### Siebel eConfigurator Version 6.x Run-Time Quote and Solution Items

For every Release 6.x quote with solution items (which means the quote was created using model products), the standard Siebel upgrade script does the following:

- If a quote solution's corresponding model product still associates with a model, the script:
  - Upgrades the solution to a top-level customizable product quote line item. The customizable product is the one that is upgraded from a model product, as described above.
  - Upgrades all quote line items under the solution to the second-level quote line items under the customizable product. The prices of the quote lines will be preserved and rolled up to the customizable product.
- If the association is removed, the script:
  - Upgrades the solution to a package product quote line item.
  - Upgrades all quote line items under the solution to the second-level quote line item under the package product. The prices of the quote lines will be preserved and rolled up to the top-level package product.

**NOTE:** This customizable product quote will be potentially reconfigurable by eConfigurator using the upgraded model, assuming the product set is maintained and the model structure is not changed after upgrade.

**NOTE:** This package product is editable but not reconfigurable by eConfigurator version 7. The solution name will be preserved in the Package column, while the model product name will not be preserved. For more information, see the section on package products in *Product Administration Guide*.

The Quote Solution applet is dropped in Siebel 7, but the concept is mostly maintained through the use of hierarchical products and package names (which replace solution names). Every quote that previously had a solution will have either a customizable product or a package product as the top-level line item in the quote, and the solution name will be maintained in the package field of the quote line item.

## Postupgrade Tasks

After you finish the automatic upgrade process, you should take the following steps to review your upgraded results before releasing to production. Some steps are recommended, and others are required. Taking these steps will make sure that your models function properly in the new environment. Due to the different model design paradigms, you may need to revise or recreate a certain amount of new configuration data to make use of the new Siebel 7 features. See [“Limitations” on page 427](#) for information about preparing data and implementation changes.

### Checking Configuration Rules (Required)

All the rules in eConfigurator version 6.x are upgraded as-is, but with “Inactive” status. Since eConfigurator version 7 uses a new model design paradigm, you should examine all your rules to verify that they still make sense. You can do this by modifying each rule and clearing the “Inactive” check box.

You also can create new rules that replace the upgraded rules. Creating new rules can have the following advantages:

- It is easier to write advanced rules in Siebel 7 using the new rules designer.
- Some rules can be consolidated or may not be needed because of new features in Siebel 7.
- Rules created in the Siebel 7 rules designer are easier to maintain.

Siebel eConfigurator version 7 provides an enhanced rules designer—Product Rule Manager—for you to create your rules. For modifying the upgraded rules, the Advanced Rule in the Product Rule Manager is particularly useful. For more information on Product Rule Manager, see *Product Administration Guide*.

There are a few considerations to keep in mind when you review and modify your rules:

- The items of the rules are specified as paths instead of Cfg IDs. Use the Advanced Rule in the Product Rule Manager and its picklist capability to replace them individually.
- Second- and lower-level categories are gone. You may need to change the references based on the new the relationship structure.
- Selection rules are no longer applicable. Assign proper relationship cardinalities and use the new UI designer features to simulate these rules.

- Some items in a Release 6.x model, such as virtual products and properties, are not supported in the Siebel 7 customizable product. You may want to use the new Siebel 7 features (such as product attributes) to reimplement the functions of these items. Following implementation of the new features, you may come back and create more rules.

The rules descriptions are preserved through the upgrade process and can serve as a good reference when you try to re-create a rule using the Product Rule Manager.

**NOTE:** All rules are upgraded with Inactive status. You should test the rules extensively before releasing your upgraded customizable products. For more information about Product Rule Manager, see *Product Administration Guide*.

### Verifying Scripts (Required)

Some Release 6.x eConfigurator scripts may no longer make sense after they are upgraded to eConfigurator version 7. Also, some event types are not supported in Siebel 7. You may have to rewrite or remove these scripts. For more information, see the sections on scripting in *Product Administration Guide*.

### Verifying Attributes Upgraded from Virtual Products (Recommended)

Every virtual product is upgraded into a new attribute and assigned to a class. The attributes can be exposed in the upgraded customizable product through Product UI Designer. You should visit these newly created attributes to see if they serve the functions that you intend.

#### *To remove or modify attributes*

- 1 Navigate to Application Administration >Class Administration view.
- 2 Find the class that corresponds to the virtual product you are verifying attributes for. The class should have the same name as the upgraded model product.
- 3 Modify or remove attributes as desired.

For more information about managing classes and attributes, see *Product Administration Guide*.

### Exposing Resources in the Run-Time User Interface (Recommended)

The resources will be upgraded, but they will not show in the selection pages in a configuration session. You may choose to use Product UI Designer to place the resource items in the appropriate UI groups. For more information, see the section on customizable products in *Product Administration Guide*.

### Rearranging Your Customizable Product Structure (Recommended)

Since the upgrade script flattens your previous model structure, you may want to rearrange your product components by creating relationships. You can do this freely within the Siebel 7 administration environment. In rearranging your customizable product structure, note the following:

- If you remove a product item from the upgraded customizable product or move it to another relationship, you may become unable to properly edit the upgraded quote using this customizable product with the configurator.
- In Release 6.x, a model can have a component product that is also associated with the model as a model product. In these instances, after upgrading, you will have a customizable product that has itself as a component product. Since this is not supported in Siebel 7, when you check the corresponding upgraded quotes eConfigurator and ePricer may not function properly. You may consider redesigning the customizable product to avoid this self-nested situation. For example, consider removing the component that is a customizable product and recreating the quote using the newer version of customizable products.

### Validating and Releasing Upgraded Customizable Products (Required)

Before releasing your customizable products to production, you should thoroughly review and test them. Make sure to follow a careful process of transitioning from development to production to verify that all models function correctly before your end users start using them.

**CAUTION:** Do not touch the upgraded quote (for example, do not click the Reprice button) before the customizable product is examined and released.

### Using the New eConfigurator Version 7 Features (Recommended)

If you have completed all the postupgrade tasks described above, you should be able to achieve a working eConfigurator environment. To implement the features new to Siebel 7, you should thoroughly understand them and you should have completed proper model planning. For more information, see *Product Administration Guide*.

This section highlights two important new features that you may want to take advantage of to help transition your models to Siebel 7. The first is the product class system and dynamic attributes. The second is the data-driven UI.

#### Using the New Product Class System and Dynamic Attributes

Do the following to take advantage of the new product class system and dynamic attributes.

- Create a class system and define appropriate attributes for the classes. Some of the previous properties can be recreated here. Set appropriate defaults.
- Assign products to the appropriate classes. These products inherit the dynamic attributes defined on the classes.
- Assign classes to the relationships created by the upgrade script. Do this as follows:
  - a Remove all the products in the relationship.  
This makes the relationship's domain type selectable.
  - b Change the relationship's domain type to Class or Dynamic Class.

- c Assign products from the desired class to the relationship.

This procedure is required if you want to write configuration rules on the relationship that refers to the dynamic attributes of all the items. For a complete set of eConfigurator version 7 feature descriptions, see *Product Administration Guide*.

- Create new rules or modify existing rules as needed, to make use of the new features and customizable product structure.

### Using the New Data-Driven UI

The upgrade script uses default UI themes and controls for all the customizable products upgraded. You may use the Siebel 7 Product UI Designer to choose or define more UI themes and elements so that your run-time product presentations are richer. For more information, see the section on customizable products in *Product Administration Guide*.

## Limitations

Upgrading to eConfigurator version 7 happens with the following limitations:

**Favorites.** Favorites are not upgraded because while the Favorites function is preserved in Siebel 7, it is implemented differently. The Release 6.x Favorites data will not upgrade to Siebel 7 automatically. You may need to recreate the Favorites data.

**Virtual products.** Virtual products are upgraded as attributes. Some functionality may be lost. You may need to define attributes or expand the upgraded attributes' domains to replace the functions of Release 6.x virtual products. You can reimplement the functionality by modifying the customizable products that contain the associated attributes. An alternative is to use Siebel eAdvisor to replace your use of virtual products if a comprehensive needs-analysis function is required.

**Configuration rules.** Your rules need to be revisited individually. To make modifications, use the Advanced Rule and Assisted Advanced Rule templates in the Product Rule Manager. Some rules will become invalid with the upgrade.

**Properties.** Properties are not upgraded. You may need to define attributes to replace them.

**Scripts.** Upgraded scripts need to be rewritten. The event types are different.

**Shared models.** Shared-model structure is redeployed in every model in which the shared model is included. The definition of each shared model becomes part of the parent model, and so the structure of the parent model is flattened before upgrade. If you want to retain the shared model approach, you must upgrade the shared model as a customizable product. Then you add it to a relationship within the parent customizable product. Doing this may prevent recustomizing upgraded quotes that contain the parent customizable product.

**Quotes.** Upgraded quotes may be read improperly by Siebel eConfigurator if you choose to remove products from the upgraded customizable product or move them to a different relationship.

**Tree-control UI.** The tree-control user interface is not available in Siebel 7. It is replaced by the more flexible data-driven UI (templates and controls).

## Case Study

This section provides a case study on how to plan and execute your upgrade. For additional detailed case studies, please contact the Siebel ISS Competency Center.

### Background

ABC Computers is a computer retail company that uses eConfigurator version 6.x to maintain its product offerings. ABC defines three models:

- "Desktop Model"
- "Laptop Model"
- "Printer Model"

“Printer Model” is used as a shared model in “Desktop Model.” Figure 10 shows the model structures of “Desktop Model.”

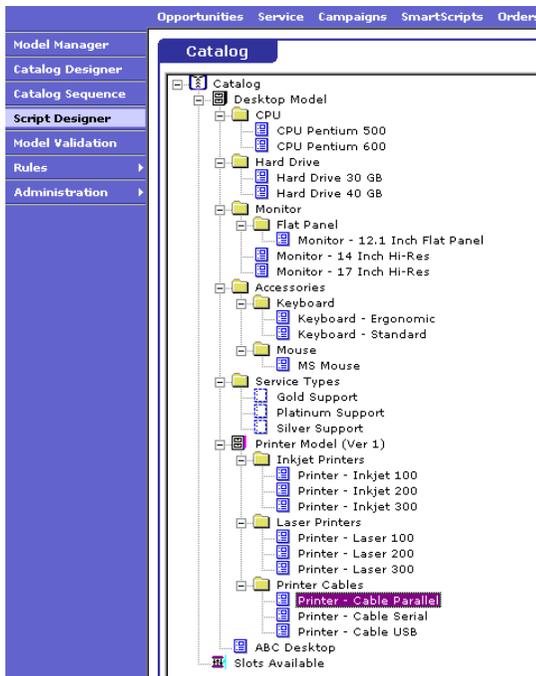


Figure 10. Release 6.x Desktop Model

Figure 11 shows the configuration rules, resources, and linked items defined for model “Desktop Model.”

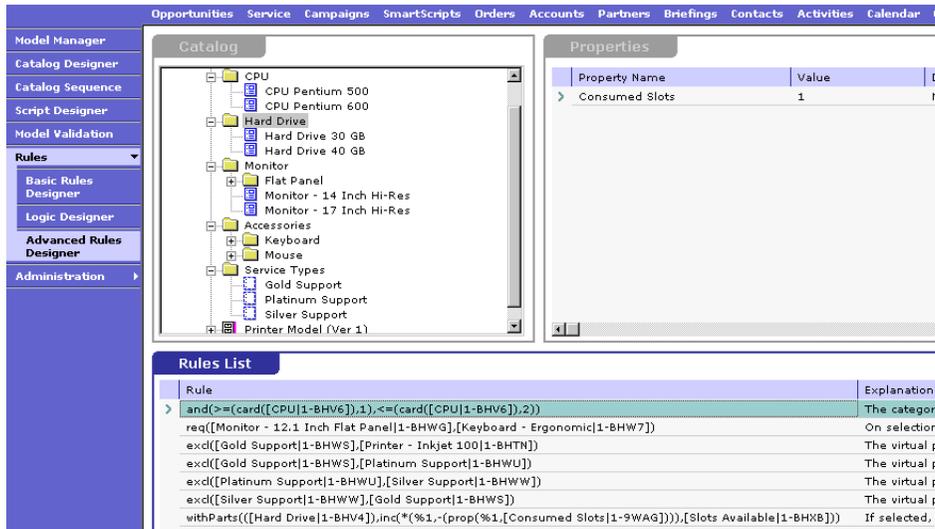


Figure 11. Release 6.x Desktop Model Rules

ABC also defines two model products:

- “ABC Desktop”
- “ABC Laptop”

These model products associate with “Desktop Model” and “Laptop Model” respectively. As shown in Figure 12, there is no model product associated with “Printer Model.”

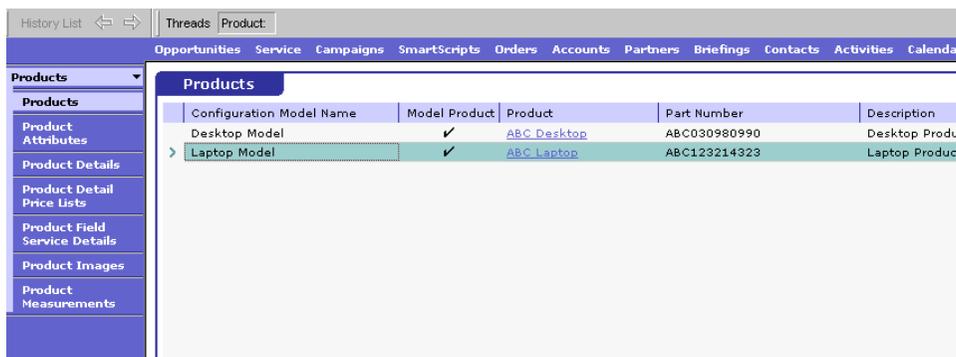


Figure 12. Release 6.x Model Products

ABC sales people use these models (and products) to create a quote (“XYZ Quote”) that has two solutions (“XYZ Desktop” and “XYZ Laptop”), as shown in Figure 13. These two solutions use the two model products “ABC Desktop” and “ABC Laptop.”

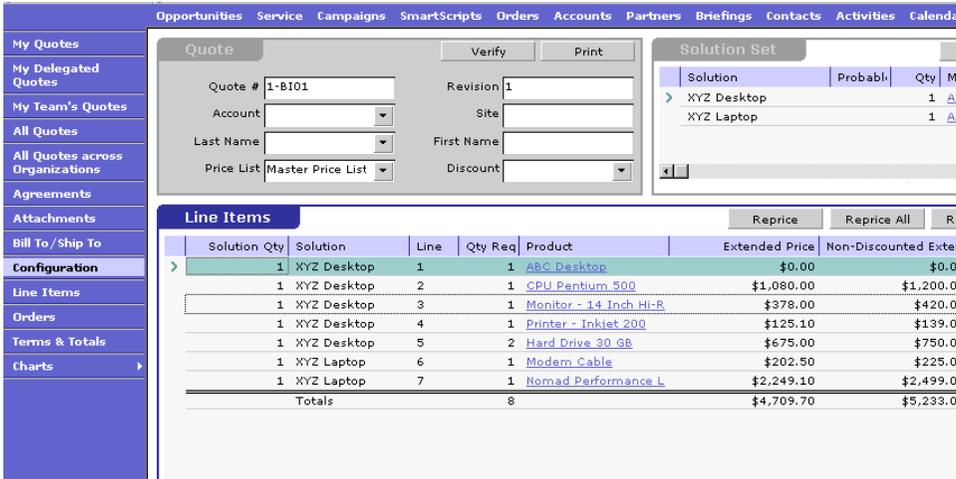


Figure 13. Release 6.x Quotes

## Postupgrade: Verify Model Associations

This preupgrade preparation completes the task of “[Verifying the Model Associations in the Product Table \(Required\)](#)” on page 414.

- 1 Assume ABC Computers is going to design a new model in Siebel 7 for their laptop products. ABC can remove the model association (Laptop Model) of product “ABC Laptop.” ABC has decided to forego the reconfiguration capability of solution “XYZ Laptop.”

- There is no model product associated with model “Printer Model.” The upgrade script will add the contents of Printer Model to XYZ Desktop but will not create a separate Printer Model customizable product. To make Printer Model a customizable product after the upgrade, ABC must create a model product and associate Printer Model with it, as shown in [Figure 14](#).

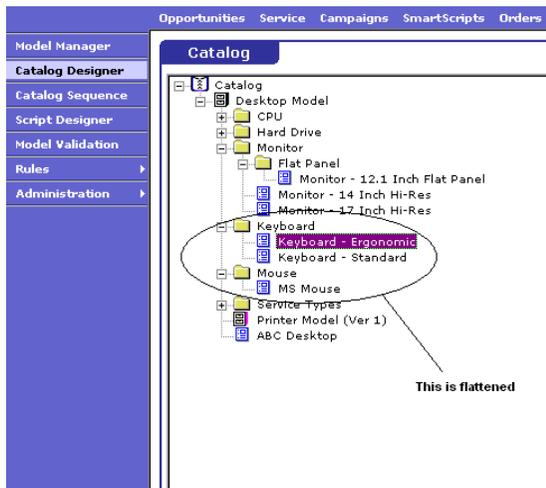
Configuration Model Name	Model Product	Product	Part Number
Desktop Model	✓	<a href="#">ABC Desktop</a>	ABC030980990
		<a href="#">ABC Laptop</a>	ABC123214323
> Printer Model	✓	<a href="#">ABC Printer</a>	ABC048394339

Figure 14. Release 6.x Model Product with Shared Model

This preupgrade preparation completes the task to “[Review and Adjust eConfigurator Version 6.x Models \(Recommended\)](#)” on page 416.

## Postupgrade: Review and Adjust Models

- ABC flattens its model structure. The figure below shows the flattening of the second-level categories. ABC leaves other structures intact for the upgrade script to do the flattening.



- ABC records the properties.

- 3 ABC examines their virtual products. In this example, the model has three virtual products for specifying the service types. To avoid creating three attributes, ABC removes two virtual products and keeps only one. If there are rules related to these virtual products, they will become invalid. Figure 15 shows the Release 6.x virtual products used as service types.

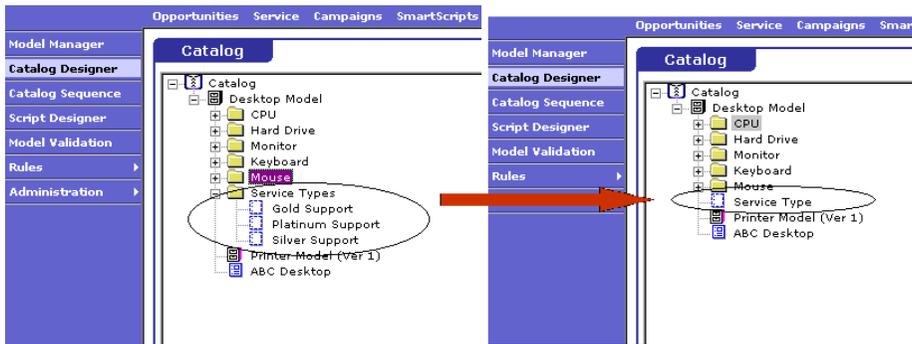


Figure 15. Release 6.x Virtual Products

## Postupgrade Review

Following the upgrade process, ABC will have:

- Two new classes: “ABC Desktop” and “ABC Printer.” See Figure 17 on page 434.
- One new attribute: “Service Type,” defined for class “ABC Desktop.”
- Two new customizable products: “ABC Desktop” and “ABC Printer,” assigned to classes “ABC Desktop” and “ABC Printer” respectively.

- ABC Printer shared model node becomes a relationship in ABC Desktop. All the products in this shared model are redeployed. (See [Figure 16.](#))

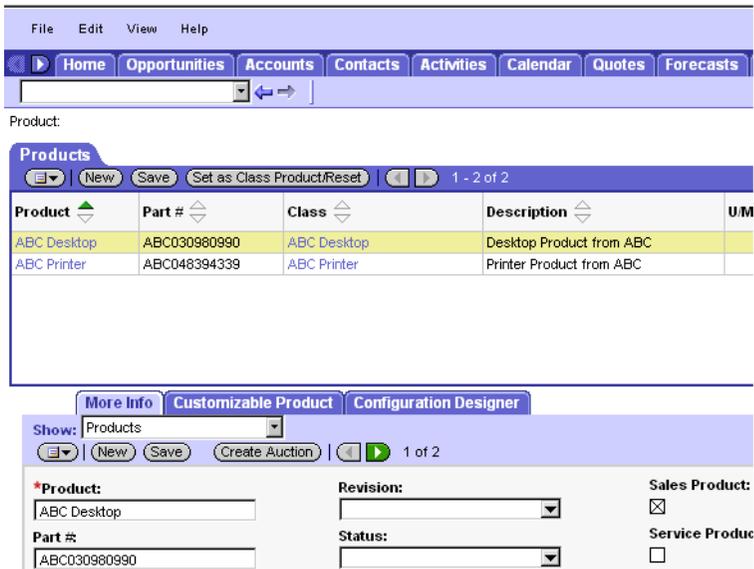


Figure 16. 7.0 Customizable Products

Figure 17 shows ABC's two new classes and the new Service Type attribute.

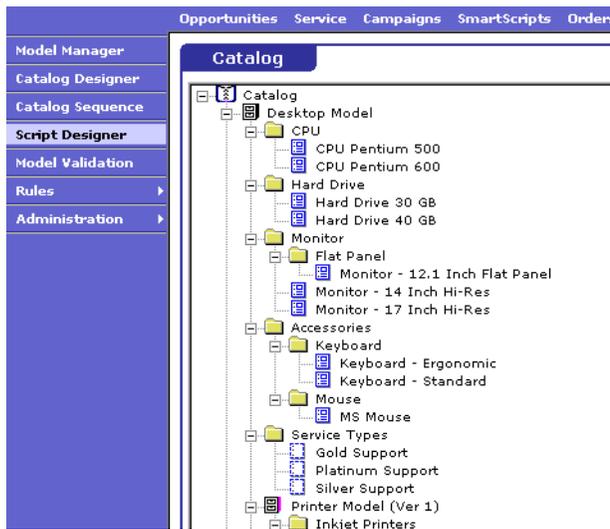


Figure 17. 7.0 Classes

The “ABC Desktop” structure is shown in Figure 18.

The screenshot displays the Siebel eConfigurator interface for the 'ABC Desktop' product. The top section shows product details: Product (ABC Desktop), Part # (ABC030980990), Description (Desktop Product from ABC), and Product Line. Below this are tabs for 'More Info', 'Customizable Product', and 'Configuration Designer'. The 'Products Designer' section shows a table of relationships between components.

Relationship	Domain Type	Min Cardinality	Max Cardinality	Default Cardinality	Product	Def
CPU	Class	0	1,000,00...	0		
Hard Drive	Class	0	1,000,00...	0		
Monitor	Class	0	1,000,00...	0		
Keyboard	Class	0	1,000,00...	0		
Keyboard - Ergonomic					Keyboard - Ergonom	
Keyboard - Standard					Keyboard - Standar	
Mouse	Class	0	1,000,00...	0		

Figure 18. “ABC Desktop” Structure

Following the upgrade process, ABC will also have:

- One quote, “XYZ Quote” with two top-level quote line items: “ABC Desktop” and “Package,” with “XYZ Desktop” and “XYZ Laptop” in the Package column to preserve the solution names. These two quote line items have their corresponding second-level quote line items upgraded from the previous solutions.
  - Quote line “ABC Desktop” can be reconfigured once the corresponding customizable product “ABC Desktop” is examined and released.
  - Quote line “Package” is now a package product and its component products can be manually edited.

## Postupgrade Review

- 1 This postupgrade review starts by walking through the task of [“Checking Configuration Rules \(Required\)”](#) on page 424.

All rules are marked “Inactive” during the upgrade process. ABC examines and activates every rule. Here are some examples of rule conversions.

Quantity rule prior to upgrade:

```
chkqty(and(>=([Monitor|1-BHV8], 1), <= ([Monitor|1-BHV8,2], 2)))
```

The category Monitor has a minimum count of 1 and a maximum count of 2.

Quantity rule converted:

On the upgrade relationship Monitor set the cardinality min = 1 and max = 1

Selection rule prior to upgrade:

```
and(>=(card([CPU|1-BHV6]), 1), <=(card([CPU|1-BHV6]), 1))
```

The category CPU has a minimum selection of 1 and a maximum selection of 1.

Selection rule converted:

Do not activate converted rule. Instead, go to the Product UI Designer and choose the Combo Box with Price and Quantity as the UI control for this relationship. This UI control limits the user to selecting only one item from the relationship but does not restrict the quantity the user can select.

Require rule prior to upgrade:

```
req([Monitor-12.1 Inch Flat Panel|1-BHWG],  
[Keyboard-Ergonomic|1-BHW7])
```

On selection, the Product Monitor-12.1 Inch Flat Panel will require the product Keyboard-Ergonomic.

Require rule converted:

Use the Advanced Rule Template to modify this rule as follows:

```
req(@.[Monitor]([12.1 Inch Flat Panel]), @.[Keyboard]([Ergonomic]))
```

- 2 Next, ABC considers the task of [“Verifying Scripts \(Required\)” on page 425](#). Since event types and the script set are different in Siebel 7, ABC makes sure all scripts remain effective and they redesign some, as necessary. All scripts are preserved in Configuration Designer >Script Designer view.
- 3 Next, ABC Computers moves on to the task of [“Verifying Attributes Upgraded from Virtual Products \(Recommended\)” on page 425](#). ABC has only one virtual product “Service Type” being upgraded to an attribute in class “ABC Desktop.” ABC can see and expose this attribute for product “ABC Desktop.” But to make it useful, ABC redefines the attribute to have the original three service types in Application Administration > Class Administration view.

Class: ABC Desktop > LOV:

List Of Values - Type  
1 - 1 of 1

Type	Translate
Service Types	

(New) (Query) 1 - 3 of 3

Language Independent Code	Display Value	Language Name	Active
Gold Support	Gold Support	English-American	✓
Platinum Support	Platinum Support	English-American	✓
Silver Support	Silver Support	English-American	✓

- 4 Now, ABC is ready to tackle the task of [“Using the New Data-Driven UI” on page 427](#). ABC chooses to use the default UI theme and controls. To customize the UI, ABC can refer to *Product Administration Guide*.

File Edit View Help

Home Opportunities Accounts Contacts Activities Calendar Quotes Forecasts Revenues Comp

ABC Desktop

Description: Desktop Product from ABC  
Total Price:  
Messages:  
[Save] [Cancel] [Done]

Service Type  
[Change]

CPU

Item	Qty
CPU Pentium 500	1

[AddItem]

Hard Drive

Item	Qty
Hard Drive 30 GB Hard Drive 40 GB	

[AddItem]

- 5 Next, ABC Computers moves on to [“Validating and Releasing Upgraded Customizable Products \(Required\)” on page 426](#). ABC uses the Validate button in Product Administration > Customizable Product > Product Versions view to validate the customizable product “ABC Desktop” before it can be used to configure more quotes or reconfigure the existing quote “XYZ Desktop.”

Later, ABC may adopt more Siebel 7 new features to this customizable product for more complex functionality.

## Approach 2: Modifying Your Existing Implementation

Since eConfigurator version 7 introduces a new model design paradigm and a set of new features, you may find that it is best to reimplement your existing configuration models and your product maintenance process.

This section details a series of processes that are commonly used for planning and implementing product definitions and configuration data. The implementation of eConfigurator version 7 differs greatly from its version 6.x predecessor in order to put the new Siebel 7 features to good use.

For more details about eConfigurator version 7 functionality, see *Product Administration Guide*.

**NOTE:** You may contact Siebel Professional Services or a certified Siebel partner to assist in your implementation of new models. All involvement on the part of Siebel Professional Services is available on a time-and-materials basis.

### Overview

To manually reimplement your models, you need to first understand the new Siebel 7 features. Then you should analyze your Release 6.x models and determine what kind of data can be carried over and what can be reimplemented by using the new features. Because [“Approach 1: Using the Standard Upgrade Tool” on page 412](#) describes the procedures that make use of most of your Release 6.x model data, it is reasonable to assume that if you choose to follow Approach 2, you will want to revamp most of your product definitions to make full use of the new features. The sections that follow outline a set of suggested processes for you to plan and implement your new Siebel 7 configuration environment.

**NOTE:** The new models you created in the Siebel 7 environment are generally not usable in reading your upgraded quotes.

### Analyzing Your Release 6.x Models

It is useful to have your existing Release 6.x models handy, but it is not necessary. You may decide to create Siebel 7 customizable products to simulate your existing models, or you may create very different structures for different models. In general, a new customizable product can contain most functionality that used to be provided through a model.

## Designing Your New Class System

You can use the new class system to logically classify your products. Another important function of the classification system is to hold and maintain product attributes. The Siebel application creates certain numbers of classes and attributes during the upgrade process (such as classes for model products, and attributes for virtual products). Still, since you are choosing to reimplement your models, it is recommended that you define a class system from the ground up.

The best way to define a class system is to think in terms of product attributes. Plan to group a set of products into a class if they share the same attributes. Then define subclasses which inherit the same attributes and which may have more specific attributes.

## Designing Dynamic Attributes

You can define your attributes in the process of defining your class system. But after the class system is stabilized, it is recommended that you review your attribute definitions and consider adding new ones. If you create effective attributes and attach them appropriately (for instance, by putting the more common attributes at higher class levels), it is easier to maintain your product definitions.

Most of the eConfigurator version 6.x properties and virtual product functions can be implemented using Siebel 7 dynamic attributes. If you define attributes properly for use with new customizable products, you can make use of the attributes in your configuration sessions.

## Assigning Products to Classes

You can assign products to classes so as to inherit the attributes defined for the classes. This is applicable both for simple products and for customizable products. For customizable products, you can define proper UI elements to present attribute selection in the run-time environment.

## Defining Customizable Products

Your understanding of the customizable-product concept will allow you to replicate your existing model information by turning some upgraded products into customizable products, or by creating new ones. The Siebel 7 customizable product feature provides many Siebel 7 model capabilities such as rules, resources, and scripts. At the same time, the feature remains product-centric in definition.

When defining a new customizable product, first consider the customizable product structure, that is, the relationships involved. By using relationship definitions that are built in a bill-of-materials fashion, you can define simple or complicated bundles that have multiple layers of product components.

## Defining the UI

After you define the structure of a customizable product, you can use the Siebel 7 Product UI Designer to determine the look and feel of your product in the run-time configuration session. Start with the standard UI templates and then design your own.

## Defining Rules

You can use the new Product Rule Manager to define business rules for your customizable products. With the new features (such as attributes and product relationships) and the new design paradigm, it is possible that the number of rules you need to define is now much smaller. Examine the rules in your existing models and selectively apply them to their corresponding customizable products.

## Defining Resources, Linked Items, and Scripts

Resources and linked items work the same way in Siebel 7 customizable products as they worked in Release 6.x models. The set of script events is revised and made simpler in Siebel 7. Your need for writing complicated scripts may decrease because of the new features.

## Defining Pricing

The products and their prices are upgraded as they were in Release 6.x. However, with the new Siebel 7 pricing features, such as attribute-based pricing and bundle pricing, you can create more complex pricing conditions for your Siebel 7 customizable products. Integration with ePricer is necessary in some cases. For more information, see *Pricing Administration Guide*.

# Upgrading from Release 5.x Product Configurator to eConfigurator Version 7

The process of upgrading from Release 5.x to Siebel 7 is identical to the process outlined in [“Upgrading from eConfigurator Version 6.x to eConfigurator Version 7” on page 412](#), because the standard Siebel upgrade process includes the necessary Release 5.x to Release 6.x migrations. The preupgrade and postupgrade tasks are very similar to those outlined in [“Upgrading from eConfigurator Version 6.x to eConfigurator Version 7.”](#)

The only difference in upgrading from Release 5.x is that to preserve or remove model product association, check the “Model Product” and “Configuration File” fields in the Product Administration view. There is no “Model Name” field in Release 5.X.

# Index

## Numerics

### 7.5.2

- 6.x, tables modified during release (table) 391
- seeded tables during upgrade (table) 375

## A

### access control

- about 134
- S\_CONTACT and S\_ORG-GROUP, about multi-org visibility for 134

### aggregate categories (new), reviewing 272

### AIX memory allocation segment space, securing 154, 198

### applet functionality, reviewing 263

### applets

- applet models, about and list of 252
- customizations, migrating to applets and views 256
- migrating applets to release 7.x 249
- new applet migration, reviewing 259
- scripts, about migrating to release 7.x 281

### Application scripts

- 7.x, about migrating to 281

### attachments

- file system attachments, updating 311

### audience for guide 118

## B

### binary sort order

- Microsoft SQL Server, verifying database was created using 175

### business component definitions, changes to party model 126

### Business Component scripts

- 7.x, about migrating to 281

### business components

- visibility properties, changing 295

## C

### call report attachments

- file names, updating 334
- updating (procedure) 334

### categorization

- about and benefits 135
- supporting data migration, differences from

- 6.x to 7.x 135

### Cell Center, locating user logins that require resolution of duplicates 340

### client configuration, migrating to Web client

- about 249
- applet and view models 252
- applet migration, reviewing 259
- applets and views, migrating, about and wizards 249
- client-side interfaces, migration tasks 283
- hidden properties, exposing 255
- scripts, analyzing for migration 285
- scripts, migration tasks 281
- views, remigrating 260

### client-side interfaces

- outbound COM interfaces 283

### clustered indexes

- considerations for 135
- rebuilding 178

### configuration file

- ePharma configuration file, renaming 341

### configuration parameters, about and table 177

### copied objects, automatic upgrade of

- upgrade ancestor, choosing 95

### custom database schema

- license keys, adding new 217

### custom objects, remapping opportunity product migration objects 307

### Customer Repository, preparing for 6.x merge

- migrating strings, merging labels, fields, and applet web templates 245
- Repository Preparation Wizard, about running 245

### Customer Repository, preparing for merge

- inheriting upgrade behavior, about 94
- New Customer repository, preparing to import 214, 240
- upgrade inheritance, configuration steps 227

## D

### data migration

- categorization, differences from 6.x to 7.x 135
- party model, support of 125

### database server configuration, verifying

- Oracle database server configuration 170
- Siebel database expected growth during upgrade (table) 128
- database sort order, verifying**
  - about 123
  - IBM DB2 UDB, verifying database created using identity sort 158
  - Microsoft SQL Server, verifying sort order on 175
- database, preparing for upgrade**
  - AIX memory allocation segment space, securing 154, 198
  - binary sort order, verifying database created using 175
  - database sort order, verifying 123
  - Microsoft SQL Server, verifying sort order on 175
  - mobile and dedicated users, preparing for upgrade 187
  - New Customer Repository, preparing to import 214, 240
  - RDBMS software, upgrading 121
- dedicated users, preparing for upgrade** 187
- .dll files, about creating and copying to server** 166
- E**
- eChannel, about preparing for upgrade** 199
- Encryption Upgrade Utility**
  - RC2 encryption, about upgrading to 132
- ePharma configuration file, renaming** 341
- F**
- file system**
  - attachments, updating 311
- Forecasting data, preparing for upgrade** 199
- G**
- Generate Reporting Relationships button, exposing** 318
- global deployment, postupgrade tasks**
  - global time zone, about setting up environment to support 319
  - global time zone, enabling after an upgrade 319
  - Unicode, migrating to 130
- grid-based applets, reviewing** 265
- guide**
  - audience for 118
  - naming and typographical conventions 33
- H**
- hidden properties, exposing** 255
- I**
- IBM DB2 UDB database**
  - .dll files, about creating and copying to server 166
  - stored procedures and user-defined functions, installing 166
- IBM DB2 UDB, postupgrade tasks**
  - P1 index, reorganizing 311
  - statistics, running 311
- IBM DB2 UDB, preparing database for upgrade**
  - 16-KB temporary tablespace, creating 161
  - 32-KB temporary tablespace, creating 161
  - clustered indexes, considerations for 135
  - database configuration parameters, about and table 160
  - DB2 permission, verifying on AIX and Solaris 160
  - DB2 UDB Application Development Client, verifying installation of 164
  - deleting old stored procedures 166
  - identity sort order, verifying database created using 158
  - instance owner permission, verifying for AIX and Solaris 161
  - RDBMS software, upgrading 121
  - sort order, verifying on 158
  - stored procedure code, copying and installing 166
  - stored procedures and user-defined functions, about 166
- ICL, about** 100
- identity sort order, verifying database created using** 158
- Incorporate Custom Layouts, about indexes** 100
  - clustered indexes, rebuilding 178
- inheriting upgrade behavior, about** 94
- item identifier, Web template** 279
- L**
- license keys**
  - custom database schema, adding new keys 217
- log files**
  - archiving manually 226
  - SiebSrvr log files, about saving 217

**M****Microsoft SQL Server**

- clustered indexes, rebuilding 178
- configuration parameters, about and table 177
- database sort order, verifying 175
- Siebel Software Configuration utility, changing language utility runs 204
- temporary database space, about 176

**migrating, client configuration to Web client**

- about 249
- applet and view models 252
- applet migration, reviewing 259
- applets and views, migrating 249
- client-side interfaces, migration tasks 283
- hidden properties, exposing 255
- scripts, analyzing for migration 285
- scripts, migration tasks 281
- views, remigrating 260

**mobile users, preparing for upgrade 187****multilingual deployment, installing additional languages 129****multilingual seed data**

- Siebel Database Server for DB2, installing 129

**multi-value group shuttle applets, reviewing 268****N****naming conventions 33****O****objects**

- descendents or copies, viewing 227
- opportunity product migration, remapping objects 307

**Oracle database**

- server configuration, before upgrading tasks 170
- sort order, verifying on 169
- upgrading, steps before upgrading 170

**P****party model**

- about 124
- business component definitions 126
- data migration, support of 125
- data model, changes to (diagram) 125
- obsolete tables in 7.x 125

**picklist visibility properties, changing planning 296**

- Configurator 149

- Employee Relationship Management (ERM), about 140
- guidelines 118
- handheld devices 148
- Resonate Central Dispatch 149
- Workflow Designer 147

**postupgrade tasks**

- asset-based ordering 331
- call report attachments, updating 334
- dock object and rule definitions, verifying are correct 308
- ePharma configuration file, renaming 341
- file system attachments, updating 311
- Generate Reporting Relationships button, exposing 318
- IBM DB2 UDB, reorganizing the P1 index 311
- inbound workflows, updating 336
- log files, manually archiving 226
- previously unused tables now used in 7.x, examples of 305
- Purchase Orders 330
- quotes 333
- reporting relationships, generating (procedure) 318
- reporting relationships, generating, about 318
- seeded workflows, incorporating prior customizations into 336
- seeded workflows, using 336
- statistics, running 311
- symbolic string model, upgrading to 302
- tables not used in 7.x, examples of 305
- Transaction Amount field in Payment lines, updating 330
- Universal Inbox 329
- user logins, locating that require resolution of duplicates 340
- visibility modes, setting for access control 295

**preparing repository, 6.x upgrade 245****pre-upgrade tasks**

- AIX memory allocation segment space, securing 154, 198
- binary sort order, verifying database created using 175
- database sort order, verifying 123
- Microsoft SQL Server, verifying sort order on 175
- mobile and dedicated users, preparing for upgrade 187
- New Customer Repository, preparing to import 214, 240
- RDBMS software, upgrading 121

**production environment upgrade**

- development environment, upgrading
  - without 217
- ePharma configuration file, renaming 341
- global time zone, about operating in 319
- New Customer Repository, preparing to import 214, 240
- preparing production upgrade on UNIX 209
- Transaction Amount field in Payment Lines, updating 330

## Q

### quotes

- upgrade, preparing for 200

**Quotes data, preparing for upgrade** 200

## R

### RC2 encryption, upgrading to

- about 132

**RDBMS software, preparing for upgrade** 121  
**release 7.5.2**

- 6.x, tables modified during release (table) 391
- seeded tables during upgrade (table) 375

### reporting relationships

- about generating 318
- Generate Reporting Relationships button, exposing 318
- generating (procedure) 318

### repositories, upgrading

- about 89
- EIM temporary columns, generating 237
- Incorporate Custom Layouts (ICL) 100
- performing merge 230
- postmerge utilities, about 97
- renaming 203
- upgrade behavior, inheriting 94

### repository merge results

- determining success 235
- reviewing conflicts 238
- reviewing deleted objects 299
- reviewing obsolete objects 300

### Repository Preparation Wizard, about

- running 245

**rich text controls, revising** 270

**RTCs, revising** 270

## S

**S\_CONTACT, multi-org visibility for** 134

**S\_LIT files, upgrading to S\_CB\_ASSET\_VER files** 311

**S\_ORG\_GROUP, multi-org visibility for** 134

**S\_SRC\_PAYMENT table, about columns added** 188

### scripts

- migrating to release 7.x 281
- migration, analyzing for 285

**seed data, installing multilingual seed data** 129

**shuttle applets, reviewing** 268

**Siebel Application Integration (EAI), about using SAP and upgrading** 66

**Siebel Business application, upgrading**

- multilingual deployment, installing additional languages 129

### Siebel data model

- access control, about 134
- categorization, about and benefits 135
- party model, business component definitions 126
- party model, changes to data model (diagram) 125
- party model, data migration in support of 125
- party table, about 124
- party table, obsolete tables in 7.x 125
- S\_SRC\_PAYMENT Table, columns added to 188

**Siebel Database Schema, upgrading**

- restarting upgrade 211

### Siebel Database Server software

- multilingual deployment, installing additional languages 129

**Siebel eChannel, about preparing for upgrade** 199

### Siebel Financial Services

- application tables, preparing for upgrade 201
- call report attachments, updating 334
- call report attachments, updating file names 334
- household data integrity, verifying 182
- opportunity product migration 307

**Siebel Life Sciences, renaming ePharma configuration file** 341

### Siebel Marketing

- hidden 6.x fields 373
- obsolete 6.x views 372
- obsolete 7.x business objects 363
- obsolete 7.x data 357
- obsolete 7.x views 364

**Siebel Purchase Orders, updating**

- Transaction Amount field 330

**Siebel Quotes, about preparing upgrade for** 200

**Siebel Repository, preparing for upgrade**

- development environment repositories, preparing 203

dock objects and visibility rules,  
recording 154

### **Siebel Software Configuration utility**

language, changing 204  
The Siebel Upgrade Wizard, about 77

### **Siebel Upgrade Wizard**

about 77

### **Siebel Workflow**

inbound workflow, updating 336  
seeded workflows, incorporating prior  
customizations into 336  
seeded workflows, using 336

### **SieSrvr log files, about saving** 217

### **sort order**

binary sort order, verified database created  
using 175  
Microsoft SQL Server, verifying on 175

### **statistics, running** 311

## **T**

### **tables**

6.x, tables modified during release  
(table) 391  
seeded tables during upgrade (table) 375

### **tables, preparing for upgrade**

clustered indexes, considerations 135  
DB2 UDB long columns, identifying for  
truncation 121, 123, 128, 152, 153,  
165, 170, 187, 214, 240

### **Tools repository, renaming** 203

### **Transaction Amount field in Payment lines, updating** 330

### **typographical conventions** 33

## **U**

### **Unicode**

migrating to, about and procedure 130  
migration, planning considerations 131

### **upgrade inheritance, configuration steps** 227

### **upgrade, planning for**

resources 115

### **upgrading ancestor, choosing an** 95

### **upgrading inheritance**

descendent or copies of an object,  
viewing 227  
parent to descendents, propagating changes  
from 96

## **V**

### **view tabs, verifying display** 276

### **views**

customization, migrating to applets and  
views 256  
migrating to release 7.x 249  
remigrating views 260  
view models, about and list of 252  
visibility properties, changing 297

### **visibility filters, reviewing** 273

### **visibility modes, setting for access control**

about and Catalog visibility 295  
Auto Query Mode, picklists that start in (list  
of) 296  
business component visibility properties,  
changing 295  
business components, visibility type used (list  
of) 295  
picklists visibility properties, changing 296  
views visibility properties, changing 297  
views, visibility type used (list of) 297

## **W**

### **Web Client Migration Wizard**

customizations, migrating to applets and  
views 256

### **Web client scripting, about in release 7.x** 281

### **Web template item identifier** 279

### **workflows**

inbound, updating 336  
seeded, incorporating prior customizations  
into 336  
seeded, using 336

