



Upgrade Guide for DB2 UDB for z/OS

Version 7.8, Rev. E
May 2012

ORACLE®

Copyright © 2005, 2012 Oracle and/or its affiliates. All rights reserved.

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

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

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

U.S. GOVERNMENT 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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

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

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Chapter 1: What's New in This Release

Chapter 2: About Upgrade Topics

- How the Upgrade Topics are Organized 21
- About Topic Applicability 21
- About Terms Used in Upgrade Topics 22
- About File Paths and Commands in Upgrade Topics 22

Chapter 3: How the Upgrade Works

- About Supported Upgrade Paths 25
- About Using Siebel Expert Services 26
- About Upgrade Environments 26
- About the Upgrade Process 29
- About Midtier-Centric and Mainframe-Centric Upgrades 34
- About the Database Server Configuration Utility 37
- About the Siebel Upgrade Wizard and Driver Files 40
- Job Flow of a Production Database (Mainframe-Centric) Upgrade 43
 - Key Members in a Mainframe-Centric Upgrade 47
 - About JCL Job Generation 47
- About the Override File 48
- About the Repository Merge 49
- About the Siebel Database Server 52

Chapter 4: How to Perform the Upgrade

- Roadmap for Performing the Upgrade 53
 - Roadmap for Upgrading from Siebel Financial Services 6.2.1 and Siebel 7.x 54
 - Roadmap for Upgrading Without a Development Environment 54
- Process of Planning the Upgrade 55
- Process of Upgrading a Development Environment from the Midtier 55

Process of Upgrading a Development Environment from the Mainframe 62
Process of Upgrading a Production Test Environment 70
Process of Tuning Upgrade Performance 75
Process of Upgrading a Production Environment 77

Chapter 5: Database and UI Upgrade Planning

Important Upgrade Planning Resources 83
Best Practices for Doing Your Upgrade 85
Planning Your Upgrade 87
 Planning Changes to the Physical Layout of the Schema 88
 Testing Before a Production Upgrade 90
 Considering Code Page Support 90
 Executions of Jobs Using Siebel-Scheduled Mode or Vendor-Scheduled Mode 91
 Source and Target Database Planning 91
 Obtaining Required Software and Hardware 92
 Obtaining Required IBM Utilities 92
 Obtaining Required Security Privileges 92
 Planning Backup and Recovery Stages 93
 Creating a Schedule for the Upgrade 94
About Estimating Database Size 94
Upgrading Your DB2 Software 95
About Multilingual Deployments 95
Upgrading to RC2 or AES Encryption 96
New Upgrade Features 96
About User Interface Changes 97
About Upgrading Access Control 98
About the Party Model 99
About the New Column Added to S_SRC_PAYMENT 101
About Database Clustered Indexes 102
About Migrating HTML Attachments to Base Tables 102

Chapter 6: Application Upgrade Planning

Upgrade Planning for Address Data Migration 103
Upgrade Planning for Siebel Employee Relationship Management (ERM) 104
Upgrade Planning for Siebel Marketing 105

Upgrade Planning for Siebel Workflow Designer	110
Upgrade Planning for Handheld Devices	111
Upgrade Planning for Resonate Central Dispatch	112
Upgrade Planning for String Translation	112

Chapter 7: Basic Database Preparations

Preparing Tables and Views for Upgrade	115
Preparing Custom Indexes for Upgrade	116
Exporting Interface Table Data	117
Preserving Dock Objects and Visibility Rules	117
Verifying Database Server Configuration	117
Securing AIX Memory Allocation Segment Space	118
Creating Storage Groups	118
Preparing the Storage Layout of the Schema	119
Modifying the Storage Control File	120
About Extracting the Storage Control File	121
Extracting the Storage Control File	122
Validating the Extracted Storage Control File	126
Reviewing the Storage Control File	127
Reviewing Tablespace Size	129
Backing Up the Database	131
Adding a Siebel User	132

Chapter 8: Preparing Application Data for Upgrade

Preparing Workflow Processes for Upgrade	133
Identifying Seed Data Customizations	134
Preparing Mobile User Data for the Database Upgrade	140
Setting Up Campaign Status Values for Siebel Marketing Upgrade	141
Preparing Address Data for Upgrade	143
Preparing Products for Upgrade	144
Preserving Marketing Segment Descriptions	144
Identifying and Resolving Duplicate Row IDs	145
Preparing Siebel eChannel Data for Upgrade	146

Preparing Forecasting Data for Upgrade 146
Preparing Products and Quotes for Upgrade 147
Verifying Household Data Integrity 147
Preparing Financial Services Application Tables for Upgrade 149
Preparing Siebel Configurator Data for Upgrade 149
Setting the Value of S_SRC_PAYMENT.TYPE_CD 150

Chapter 9: Preparing a Development Environment for Upgrade

Prerequisites for a Development Upgrade 151
About Moving Tables 151
Checking In Development Repository Projects 152
Preparing EIM Tables for Upgrade 153
Preparing for Intersection Table Maintenance and Deduplication 155
Determining Which Template File Was Used During an Extract or Merge 156
Renaming the Siebel Repository Using Siebel Tools 157

Chapter 10: Preparing a Production Environment for Upgrade

Prerequisites for a Production Upgrade 159
Transferring the Customized Repository and Schema Definition Files 161
Renaming the Production Environment Repository 162

Chapter 11: Upgrading the Database

Changing the Database Server Configuration Utility Language 163
Preparing to Run the Database Server Configuration Utility 164
Running the Database Server Configuration Utility Under Windows 168
Running the Database Server Configuration Utility Under UNIX 169
Starting the Siebel Upgrade Wizard 170
Stopping the Siebel Upgrade Wizard 172
Migrating Address Data from Custom Extension Columns 173
Preparing for a Non-Development-Environment Upgrade 180
Installing New License Keys After Upgrade 180

Deleting Redundant Upgrade Files 181

Chapter 12: Upgrading a Development Environment Database from the Midtier

Executing SQL Files After the First Pause 183

Resuming the Siebel Upgrade Wizard After the First Pause 185

Executing SQL Output Files After the Second Pause 186

Executing SQL Output Files After the Second Pause - Stage 1 of 2 187

Deploying Stored Procedures and User-Defined Functions 188

Executing SQL Output Files After the Second Pause - Stage 2 of 2 193

Resuming the Siebel Upgrade Wizard After the Second Pause 193

Executing SQL Output Files After the Third Pause 195

Resuming the Siebel Upgrade Wizard After the Third Pause 195

Updating Statistics 197

Synchronizing the Schema 198

Dedup Files 199

Migrating Custom Business Component Configurations 199

Chapter 13: Upgrading a Database from the Mainframe

Generating Development Environment Mainframe-Centric Upgrade Files 202

Transferring the dedup Files 203

Preparing the z/OS Host Environment 203

Prerequisites for Upgrade Procedures on the z/OS Host 203

Transferring JCL, SQL, and Binary Files to the z/OS Host 204

Customizing the JCL UNIT Parameter Value 205

Defining and Allocating z/OS Setup Datasets 205

Setting Up the Upgrade Environment on the z/OS Host 207

Preparing for Table Creation (PRET) on the Source Database 210

Resuming the Siebel Upgrade Wizard After the First Pause 213

Transferring Control Cards and Schema DDL Files to the z/OS Host 215

Preparing the Schema and JCL Files on the z/OS Host 216

Applying *DSNHLQ* to FTP DDL, Unload Cards, and Load Cards 216

Unpacking the Unload and Load Control Cards and Executing the DDL 217

Installing Stored Procedures on the z/OS Host 219

Preparing Siebel-Scheduled Jobs on the z/OS Host 221

Optimizing the Target Job Stream, Part 1	224
Optimizing Old-Schema Index Rebuild Jobstreams	225
Loading Log Tables on the Source Database	226
Loading Log Tables on the Target Database	227
Executing the Upgrade on the z/OS Host (Phase 1 of 2)	228
Running Unload Jobs on the Source Database	228
Running Load Jobs on the Target Database	229
Executing Index DDL and Executing Rebuild Indexes	230
Migrating Preschm Data	231
Migrating Address Data	232
Migrating eChannel Data	233
Migrating Household Data	234
Migrating Preschm Data for FINS	234
Migrating Product Configurator Data	235
Migrating Product Configurator Data for FINS	236
Migrating ISS Data	237
Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause	238
Transferring the Index Schema to the z/OS Host	239
Preparing Siebel-Scheduled JCL	240
Optimizing the Target Job Stream, Part 2	242
Executing the Upgrade on the z/OS Host (Phase 2 of 2)	243
Loading Target Siebel Log Tables	244
Dropping Old Indexes and Creating New Indexes	244
Rebuilding the Gen Primary 7.8 Indexes	246
Migrating the Gen Primary 7.8 SQL	246
Creating the New EIM Indexes	247
Generating RUNSTATS	248
Performing Intersection Table Maintenance	249
Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause	251
Restarting Failed Jobs on the Mainframe	253

Chapter 14: Reviewing the Database Upgrade Log Files

About the Database Upgrade Logs	255
Reviewing Upgrade Log Files for Errors	256
Reviewing the Log Files	257
Reviewing Log Files for Errors in Jobs Run from the Midtier	258

- Viewing the Siebel Job Log Status 258
- Running SQL in Siebel Logs 259
- Manually Archiving Upgrade Log Files 259

Chapter 15: Performing the Siebel Tools Repository Merge

- About Inheriting Upgrade Behavior 261
- About the Postmerge Utilities 264
- About the Incorporate Custom Layout (ICL) Upgrade Option 264
- Configuring Objects to Inherit Upgrade Behavior 270
- Migrating Repository Objects to the Standard UI 271
- About Backing Up the New Customer Repository or Database Schema 275
- About Reorganizing Tables Before the Repository Merge 275
- Running the Repository Preparation Wizard on Release 6.2.1 Repository 277
- Performing a Repository Merge 279
- Determining If a Repository Merge Was Successful 288
- Migrating Custom Workflows 289
- Reviewing Attribute Conflicts in the Repository Merge 291
- Reviewing Deleted Objects in the Repository Merge 292
- Reviewing Obsolete Objects in the Repository Merge 293
- Generating EIM Temporary Columns After a Repository Merge 294
- Regenerating the Repository Definition Files 295
- Deleting Unneeded Repository Files 297
- Creating a New SRF File 297

Chapter 16: Migrating Siebel 6.2.1 Customizations

- About Migrating to the Siebel Web Client 301
- About Migrating 6.2.1 Scripts 307
- About Migrating 6.2.1 Client-Side Interfaces 308
- Identifying What Will Be Migrated 309
- Exposing Hidden Properties of Applets and Views 310
- Running the Siebel Web Client Migration Wizard 311
- Reviewing Applet and View Migration 314

Running the Web Layout Wizard	315
Migrating Scripts Attached to Controls	316
Migrating Scripts Attached to Applets	318
Migrating Business Component, Business Service, and Application Scripts	321
Migrating Outbound COM Interfaces	322
Migrating Inbound COM Interfaces	323
Regenerating the Postmerge Utilities Log	326
Reviewing Customized Business Components	326

Chapter 17: Postmerge Development Tasks

Running the Postmerge Utilities	329
Troubleshooting Postmerge UI Problems	330
Verifying Business Address Applet Configuration	331
Reviewing Grid-Based Applets After the Merge	332
Reviewing UI Navigation After the Merge	333
Reviewing Multi-Value Group (MVG) Shuttle Applets After the Merge	334
Revising Rich Text Controls After the Merge	336
Reviewing New Aggregate Categories After the Merge	337
Revising Visibility Filters to Display Correctly After the Merge	338
Assigning a Category and Type to Chart Views After the Merge	339
Assigning a Category and Type to Explorer Views After the Merge	340
Setting Up Navigation to Inaccessible Detail Views After the Merge	340
Eliminating Obsolete UI Fields After the Merge	341
Reviewing UI Objects Affected by ICL After the Merge	342
Reviewing Required Fields in the UI After Merge	342
Setting Label Alignment for Text Fields	343
Assigning an Item Identifier to Web Template Items	343

Chapter 18: Postupgrade Tasks for a Development Environment

Reapplying Custom Columns to the Siebel Database	345
Deleting Duplicate EIM Mappings	347

Regenerating the Database Template File	348
Extracting Developers or Siebel Tools Clients	349
Resetting Database Server Configuration Parameters	349
Upgrading to the Symbolic String Model	349

Chapter 19: Postupgrade Tasks for a Production Environment

Upgrading Siebel Mobile and Dedicated Web Clients	351
Upgrading Regional Servers	351

Chapter 20: Postupgrade Tasks for Database and File System

Checking for Inactivated EIM Table Columns	353
Validating Dock Objects and Rule Definitions	353
Updating the File System Directory	354
Updating File System Attachments	355

Chapter 21: Postupgrade Tasks for Applications

Generating Reporting Relationships After Upgrade	360
Setting Up Your Environment to Support Global Time Zone	362
Updating Enterprise Application Integration (EAI) After Upgrade	363
Setting Visibility Modes for Access Control	364
Removing Call Center Duplicate Logins After Upgrade	367
Upgrading ERM Customized Microsite and Group News Pages	367
Migrating Course Duration Information for Siebel Training	369
Upgrading the Launch Field in Siebel Training LOV	369
Upgrading the Test Status in Siebel Training After Upgrade	370
Verifying Class and Session Times in Siebel Training	370
Upgrading ERM Approval Business Process Workflows	371
Migrating Data to the Bankruptcy Status Field	373
Upgrading File System Attachments for Siebel Financial Services Call Reports	373
Upgrading File System for Household Notes and Attachments	374
Upgrading Marketing Responsibilities	376

Reviewing Marketing Campaign Data	376
Reviewing Renamed Fields in Siebel Marketing	377
Displaying Marketing Regions	377
Revising Marketing Program Flowchart Icons	378
Setting Default Campaign Execution Options	378
Upgrading Activity Plans for Programs and Campaigns	378
Upgrading the Newsletter Offer Type	379
Configuring Universal Inbox	379
Reviewing Data Warehouse Schema Changes	380
Upgrading Siebel Purchase Orders	380
Configuring Asset-Based Ordering	381
Reviewing Address Data After Upgrade	382
Configuring Products and Quotes	383
Upgrading Attribute Pricing	386
Verifying the Upgrade to Aggregate Discounts in Pricer	386
Migrating Data to the Bankruptcy Status Field	387
Upgrading Seeded Workflows	388
Upgrading Inbound Workflows	388
Upgrading Siebel Product Configurator	389

Chapter 22: Tuning the Production Upgrade Scripts

About Tuning the Upgrade Scripts	391
Optimizing Unload and Load Job Performance	392
Adding the Statistics Clause to Load Cards	392
About Deactivating Jobs That Do Not Process Data	393

Appendix A: Tables Modified or Seeded During Upgrade

Important Schema Changes at Release 7.8	395
---	-----

Appendix B: Siebel Marketing Upgrade Reference

Obsolete Siebel Marketing Data	401
Obsolete Business Objects in Siebel Marketing	407
Obsolete or Replaced Views in Siebel Marketing	408

Appendix C: Upgrade Planning Worksheet

Team Lead Summary	417
DB2 Connect Information	417
Siebel Development Environment Information	418
Siebel Production Environment Information	419
z/OS Host System Variables Information (Mainframe Upgrade)	420

Appendix D: Columns Denormalized During Upgrade to Release 7.8

Denormalized Columns for 6.2.1 Siebel Financial Services Applications	421
Denormalized Columns for 6.2.1 Siebel eBusiness Applications	430
Denormalized Columns for 6.3 Siebel Industry Solutions	438
Denormalized Columns for 7.0.4 Siebel eBusiness Applications	448
Denormalized Columns for 7.0.4 Siebel Financial Services Applications	450
Denormalized Columns for 7.0.4 Siebel Industry Solutions	454
Denormalized Columns for 7.5.2 Siebel Industry Applications	459

Appendix E: Production Upgrade Files Generated by the Upgrade Wizard

Siebel 7.8 Upgrade Files	461
Tables Amended During PRET Unload Processing	465
PRET Members Generated By Pretedit.txt	467
Source Tables Amended During PRET Processing	468

Index

1

What's New in This Release

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev E

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

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev D

Table 1 lists changes in this version of the documentation to support Release 7.8.2 of the software.

Table 1. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. D

Topic	Description
"Process of Upgrading a Development Environment from the Midtier" on page 55	Changed each of the upgrade processes to indicate that you must upgrade third-party software before you upgrade the servers.
"Modifying the Storage Control File" on page 120	Revised topic to include additional information about the ways in which storage space can be configured.
"Deleting Redundant Upgrade Files" on page 181	New topic describing how to delete stored procedures after the upgrade is successfully completed.
"About the Database Upgrade Logs" on page 255	Revised topic to include additional detail on the Upgrade Wizard log files.
"Siebel 7.8 Upgrade Files" on page 461	Revised the table listing the 7.8 upgrade files. Added descriptions for a number of the files listed, and corrected some of the applicable upgrade paths for the files.

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev C

Table 2 lists changes in this version of the documentation to support Release 7.8.2 of the software.

Table 2. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. C

Topic	Description
"About Using Siebel Expert Services" on page 26	New topic describing when you should contact Siebel Expert Services for help and assistance.
"About Midtier-Centric and Mainframe-Centric Upgrades" on page 34	Revised topic to clarify the differences between mainframe-centric and midtier-centric upgrades.

Table 2. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. C

Topic	Description
“About the Siebel Upgrade Wizard and Driver Files” on page 40	New topic. This topic describes how the Upgrade Wizard works and the relation between it and the driver files.
“Job Flow of a Production Database (Mainframe-Centric) Upgrade” on page 43	New topic describing the main steps involved in a production database upgrade.
“Process of Upgrading a Production Test Environment” on page 70	New topic. This topic lists the steps involved in performing a production test upgrade.
“Process of Tuning Upgrade Performance” on page 75	New topic. This topic describes how to tune upgrade scripts when performing a production test upgrade.
“Considering Code Page Support” on page 90	Added new information on code page conversion issues.
“Generating Development Environment Mainframe-Centric Upgrade Files” on page 202	New topic that describes how to copy the files required to perform a mainframe-centric development environment upgrade to the DBSRVR directory on the midtier.
“Transferring the dedup Files” on page 203	New topic describing how to transfer the files generated by the Prepare for Production mode of the Database Server Configuration utility to the appropriate directory.
“Customizing the JCL UNIT Parameter Value” on page 205	New topic describing how to amend the UNIT=SYSDA parameter setting for all of the JCL generated for the Siebel upgrade before you run jobs on the mainframe.
“Preparing for Table Creation (PRET) on the Source Database” on page 210	Revised topic to include additional information on adding job cards to PRET jobs.
“Migrating Preschm Data” on page 231	Revised topic. Added additional information about the PRESCHM jobs to this topic.
“Restarting Failed Jobs on the Mainframe” on page 253	New topic describing how to restart upgrade jobs on the z/OS host.
“About Backing Up the New Customer Repository or Database Schema” on page 275	New topic describing the options available for restoring the database environment to its pre-merge state if the repository merge fails.
“About Reorganizing Tables Before the Repository Merge” on page 275	New topic listing the tables that should be reorganized before you run the repository merge.
Chapter 22, “Tuning the Production Upgrade Scripts”	New chapter. This chapter describes how to tune the upgrade scripts during the production test upgrade so they can be re-used in the production upgrade.
Appendix E, “Production Upgrade Files Generated by the Upgrade Wizard”	New appendix. This appendix provides information about the files that are generated when you run the Siebel Upgrade Wizard.

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev B

Table 3 lists changes in this version of the documentation to support Release 7.8.2 of the software.

Table 3. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. B

Topic	Description
Chapter 3, "How the Upgrade Works"	Revised chapter to include information on the Prepare for Production Upgrade step.
"Process of Upgrading a Development Environment from the Midtier" on page 55	Revised topic. Corrected procedure for upgrading the Siebel Database Schema (upgrep) and the Custom Database Schema (upgphys).
"Process of Upgrading a Development Environment from the Mainframe" on page 62	Revised topic. Corrected procedure for upgrading the Siebel Database Schema (upgrep) and the Custom Database Schema (upgphys).
"Process of Upgrading a Production Test Environment" on page 70	Revised topic. Corrected procedure for performing a production upgrade.
"Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause" on page 251	New topic added to the procedure for upgrading a database from the mainframe.

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. A

Table 4 lists changes in this version of the documentation to support Release 7.8.2 of the software.

Table 4. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. A

Topic	Description
"Important Upgrade Planning Resources" on page 83	Revised topic. SupportWeb now has an installation and upgrade portal page.
"Roadmap for Performing the Upgrade" on page 53	Revised topic. Added a Roadmap for upgrading from Release 7.8.1 to Release 7.8.2.
"Best Practices for Doing Your Upgrade" on page 85	Revised topic. Added steps for determining your upgrade path.
"About Multilingual Deployments" on page 95	Revised topic. For multilingual deployments, you must manually import language-specific repository strings and seed data after upgrep.
"Creating Storage Groups" on page 118	New topic. Seed data is overwritten when upgrading from Release 7.8.1 to 7.8.2 or later. Before upgrading, you can run a report to identify seed data customizations.

Table 4. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8 Rev. A

Topic	Description
“Setting the Value of S_SRC_PAYMENT.TYPE_CD” on page 150	New topic. S_SRC_PAYMENT.TYPE_CD is a required field. For records where this is null, set a value prior to upgrade.
“Running the Database Server Configuration Utility Under UNIX” on page 169	Revised topic. Corrected and clarified procedure steps.
“Preparing to Run the Database Server Configuration Utility” on page 164	Revised topic. For multilingual deployments, you must select the primary (base) language when you run the Database Server Configuration utility.
“Reviewing Upgrade Log Files for Errors” on page 256	Revised topic. Corrected file names of logs that can be ignored.
“About the Incorporate Custom Layout (ICL) Upgrade Option” on page 264	Revised topic: <ul style="list-style-type: none"> ■ In an ICL merge, new controls from the New Siebel Repository are not copied to applets. ■ 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.
“Migrating Custom Workflows” on page 289	New topic. If you upgrading from Release 7.7.x or 7.8.x, you must migrate custom workflows to the New Customer Repository prior to the development environment upgphys.
“Reviewing Customized Business Components”	Deleted topic from “Migrating Siebel 6.2.1 Customizations” on page 301 . This topic is no longer applicable.

What's New in Upgrade Guide for DB2 UDB for z/OS, Version 7.8

Table 5 lists changes in this version of the documentation to support Release 7.8.1 of the software.

Table 5. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8

Topic	Description
Chapter 13, “Upgrading a Database from the Mainframe”	New chapter. There are now two ways to upgrade a development environment database: from the midtier or from the mainframe. Review this chapter for information on upgrading a development environment database from the mainframe.
“Upgrade Planning for Address Data Migration” on page 103	New topic. The way addresses are stored has been changed at Release 7.8. Review this topic to determine how to handle table customizations.

Table 5. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8

Topic	Description
“Preparing Products for Upgrade” on page 144	New topic. To migrate products in workspaces, you must release them. Class products must have the orderable flag unchecked.
“Preparing Address Data for Upgrade” on page 143	New topic. You must check for duplicate row-IDs within and between S_ADDR_PER and S_ADDR_ORG before performing the upprep of the Siebel Database.
“Migrating Repository Objects to the Standard UI” on page 271	New topic. If you selected the Incorporate Custom Layout (ICL) option for the previous Release 7.x upgrade, you must return your user interface to the standard look and feel before performing the repository merge.
“About the Incorporate Custom Layout (ICL) Upgrade Option” on page 264	Revised topic. Added guidelines for deciding when to use Incorporate Custom Layout (ICL).
“Migrating Address Data from Custom Extension Columns” on page 173	Revised topic. The changes required to the ddl.ctf and preschm_sia.sql files to migrate address data in custom extension columns has been heavily updated. Review this topic carefully before migrating address data from custom extension columns.
“Running the Postmerge Utilities” on page 329	New topic. You must start the postmerge utilities manually. You can run them multiple times.
“Setting Label Alignment for Text Fields” on page 343	New topic. If you select the ICL option for the upgrade and choose “Label on Top,” you must edit main.css to obtain correct label alignment.
“Verifying Class and Session Times in Siebel Training” on page 370	New topic. You must verify class and session start times in Siebel Training after upgrade.
“Reviewing Address Data After Upgrade” on page 382	New topic. You must review address records after upgrade to eliminate duplicate and obsolete records.
“Upgrading Attribute Pricing” on page 386	New topic. You must run a business service method to upgrade attribute pricing.
“Verifying the Upgrade to Aggregate Discounts in Pricer” on page 386	New topic. You must verify bundle factors have been correctly updated to the new aggregate discounts feature.
“Important Schema Changes at Release 7.8” on page 395	New topic. Lists important schema changes in Release 7.8.
“Upgrading ERM Customized Microsite and Group News Pages” on page 367	Revised topic. Corrected error in procedure.

Table 5. New Features in Upgrade Guide for DB2 UDB for z/OS, Version 7.8

Topic	Description
Chapter 17, “Postmerge Development Tasks”	<p>All topics rewritten to improve clarity. Tools procedures added where necessary.</p> <p>Three new topics added covering “Issue 7, 8, and 9” sections of postmerge utilities log.</p>
“Migrating Data to the Bankruptcy Status Field” on page 387	<p>New topic. BK_STATUS_CD is added to S_BANKRUPTCY. This column stores bankruptcy status for use by Siebel Financial Services applications. If you have implemented a bankruptcy status field, you must migrate the data to BK_STATUS_CD.</p>

2

About Upgrade Topics

This chapter provides general information about the upgrade topics. It includes the following topics:

- [“How the Upgrade Topics are Organized”](#)
- [“About Topic Applicability”](#)
- [“About Terms Used in Upgrade Topics” on page 22](#)
- [“About File Paths and Commands in Upgrade Topics” on page 22](#)

How the Upgrade Topics are Organized

Upgrades: All upgrades.

Use the roadmaps and process topics in [Chapter 4, “How to Perform the Upgrade,”](#) to guide you through the upgrade process. These topics provide a checklist of all the steps required to complete a particular type of upgrade, in the order in which you must perform them. Each step includes a link to a topic that explains how to complete the step.

The remaining chapters of the guide are organized according to the major phases of the upgrade. Each chapter includes the specific upgrade tasks you must perform for that portion of the upgrade as well as conceptual and process information relating to those tasks.

CAUTION: Topics in the chapters may not follow the order you perform them during the upgrade and, depending on your upgrade path, all topics may not apply. You must use the roadmap for your upgrade to determine the required and optional steps and their sequence. If you do not, you are likely to experience difficulties in completing your upgrade.

About Topic Applicability

Upgrades: All upgrades.

Environments: All environments.

The applicability of topics in the book is listed at the beginning of each topic. [Table 6 on page 22](#) lists the applicability categories and their meaning.

For each topic, only the relevant categories are listed.

Table 6. Topic Applicability Categories

Applicability Category	Meaning
Upgrades	<p>Lists the upgrades to which the topic applies.</p> <p>For example, Upgrades: Releases 7.0.x, 7.5.x. means the topic applies to upgrades from 7.0.x or 7.5.x. The topic does <i>not</i> apply to upgrades from 7.7.x or later.</p>
Environments	<p>Lists the Siebel environments to which the topic applies.</p> <p>For example, Environments: Development environment only. means the topic applies only to a development environment upgrade.</p> <p>For more information on Siebel environments, see “About Upgrade Environments” on page 26.</p>

About Terms Used in Upgrade Topics

Upgrades: All upgrades.

Environments: All environments.

This guide follows several naming conventions:

- *DB2* refers to IBM DB2 UDB for z/OS.
- Current release means the currently shipping release of the Siebel Business applications.
- Release 7.x refers collectively to all versions of Siebel 7 prior to Release 7.8, for example Release 7.0.4 or 7.5.3. See the Upgrade section in *Siebel System Requirements and Supported Platforms* on Oracle Technology Network for specific information on which release numbers are meant by Release 7.x.
- The term Windows refers to all Microsoft Windows operating systems listed as supported for this release in *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.
- The term UNIX refers to all forms of the UNIX operating system supported for this release in *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

About File Paths and Commands in Upgrade Topics

Upgrades: All upgrades.

Environments: All environments.

Environment variables and path placeholders for both Windows and UNIX paths are used throughout this guide. You must enter UNIX commands in either a C-shell or Korn shell. Enter Windows commands in a Windows Command Prompt window.

Windows Paths

The following path conventions specify file system locations in topics:

- *SI EBEL_ROOT* is the absolute path of the Siebel Server installation directory. When you install a Siebel Server, the installation script queries for the installation path. The script then installs the Siebel Server in a subdirectory of this path called *siebsrvr*. For example, if you specified `C:\sea7xx` as the installation path, then *SI EBEL_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 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 *SI EBEL_ROOT* is `C:\sea7xx\si ebsrvr`, then *DBSRVR_ROOT* is `C:\sea7xx\dbsrvr`.

UNIX Paths

The following environment variables and path conventions specify file system locations in *Upgrade Guide for DB2 UDB for z/OS* topics:

- `$SI EBEL_ROOT` is an environment variable that defines the absolute path of the Siebel Server installation directory. When you install a Siebel Server, the installation script queries for the installation path. The script then installs the Siebel Server in a subdirectory of this path called *siebsrvr*. For example, if you specified `usr/si ebel` as the installation directory, then `$SI EBEL_ROOT` is `usr/si ebel /si ebsrvr`.

The definition of `$SI EBEL_ROOT` and other environment variables required for doing an upgrade are located in `/si ebsrvr/si ebenv.csh`, (`si ebenv.sh` for Korn shells). The Siebel Server installation script creates 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 this guide. It is not an environment variable and is not defined in `siebenv.csh` or `siebenv.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 `$SI EBEL_ROOT` is `usr/si ebel /si ebsrvr`, then *DBSRVR_ROOT* is `usr/si ebel /dbsrvr`.

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 filenames, 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.

3

How the Upgrade Works

This chapter provides an overview of the upgrade process, upgrade environments and the utility used to perform the upgrade. It includes the following topics:

- [“About Supported Upgrade Paths”](#)
- [“About Using Siebel Expert Services” on page 26](#)
- [“About Upgrade Environments” on page 26](#)
- [“About the Upgrade Process” on page 29](#)
- [“About Midtier-Centric and Mainframe-Centric Upgrades” on page 34](#)
- [“About the Database Server Configuration Utility” on page 37](#)
- [“About the Siebel Upgrade Wizard and Driver Files” on page 40](#)
- [“Job Flow of a Production Database \(Mainframe-Centric\) Upgrade” on page 43](#)
- [“About the Override File” on page 48](#)
- [“About the Repository Merge” on page 49](#)
- [“About the Siebel Database Server” on page 52](#)

About Supported Upgrade Paths

Upgrades: All upgrades.

Environments: All environments.

Supported upgrade paths are described in *Siebel System Requirements and Supported Platforms* on Oracle Technology Network. This guide describes how to upgrade the following installations:

- A Siebel Financial Services 6.2.1 installation to Release 7.8.
- A Siebel 7.x installation to Release 7.8.

This 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:

- Changing operating system type during an upgrade, for example changing from Windows to UNIX.
- Migrating from Siebel Industry Solutions or Siebel Financial Services applications to Siebel Business applications.

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

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

About Using Siebel Expert Services

The Siebel 7.8 upgrade process on DB2 for z/OS is designed to run in all standard Siebel implementations. It is built on the assumption that data exists in all Siebel tables and that all this data needs to be migrated.

In reality, your implementation has probably been customized to suit your business so, for example, you might not use all of the Siebel tables shipped or they might contain varying amounts of data. To accommodate this fact, the upgrade process is customizable, for example, you can eliminate upgrade jobs that run on empty tables, or you can choose to run unload jobs simultaneously.

Siebel Technical Support provides support for all standard z/OS upgrades but it does not support customized upgrades. Customizing the 7.8 upgrade scripts is a complex process and, for this reason, if you want to customize the upgrade scripts, you must contact Siebel Expert Services for assistance and approval of the changes.

This guide describes a number of upgrade customization tasks that you can perform where the assistance of Siebel Expert Services is *required*.

CAUTION: You *must* contact Siebel Expert Services before performing tasks where such help is noted as a *requirement*. If you do not, you may invalidate your support agreement.

This guide also describes tasks where enlisting the help of Siebel Expert Services is *recommended*. Failure to contact Siebel Expert Services for help with these tasks *does not* have implications for continuing Siebel support.

Where a task requires the help of Siebel Expert Services, this is indicated in the relevant topic.

About Upgrade Environments

Upgrades: All upgrades.

Environments: All environments.

The guide describes how to upgrade three environments:

- Development environment
- Production test environment
- Production environment

The upgrade process differs in each environment.

Development Environment

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

A development environment contains the following elements:

- 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.

There are two scenarios for development environment upgrades:

- **Development environment database with DB2 UDB.** If your development environment database is DB2 UDB, you must use the *Upgrade Guide* for your development environment upgrade. Then use this guide for your production environment upgrade.
- **Development environment database with DB2 UDB for z/OS.** During DB2 UDB for z/OS upgrades, the Siebel Upgrade Wizard generates the schema and output files. Your database administrator must review these files and apply them on the z/OS host.

Production Test Environment

The production test environment is where you test the upgraded release to validate its function and performance before deploying it to users. This is also where you tune the upgrade process to minimize the time required to perform your production upgrade.

By tuning the production upgrade scripts in a test environment, you 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.

This environment includes the following elements:

- 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.

Tuning the upgrade 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.

Siebel Systems provides these tools to help you transition from production test to production:

- **Siebel Application Deployment Manager (ADM).** This application 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 Mobile or Developer Web Client. 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 7](#) gives recommendations for mapping your environments to the ones described in the this guide.

Table 7. Mapping Your Environments to Siebel Upgrade Processes

Environment Description	Recommended Upgrade
The environment has the following characteristics: <ul style="list-style-type: none"> ■ It is used primarily for development with Siebel Tools. ■ The Siebel Database is a subset of your production database. ■ The environment is not used for technical support or training. Developers are usually installed as Mobile Web Clients. 	Development environment upgrade.
The environment has the following characteristics: <ul style="list-style-type: none"> ■ It is intended for testing customizations before deploying them. ■ It is where you tune your upgrade SQL files to minimize production upgrade time. ■ 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. 	Production test environment upgrade.
The environment is used for live business transactions by both local and remote users.	Production environment upgrade.

About the Upgrade Process

Upgrades: All upgrades.

Environments: All environments.

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. 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.

NOTE: In a production test and production environment upgrade, the upprep + upgphys steps are run together and there are also several additional deployment steps.

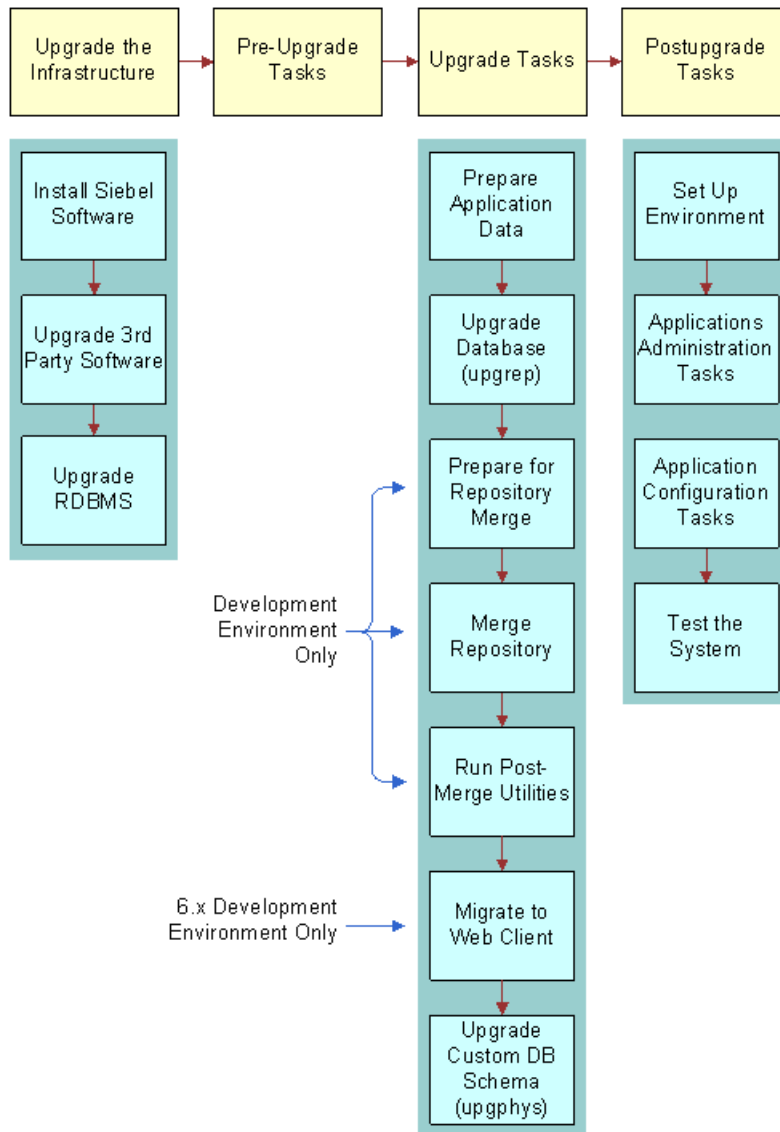


Figure 1. Flow of the Upgrade Process

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 the *Siebel Installation Guide* for the operating system you are using.

Perform Preupgrade Tasks

This phase prepares the Siebel Database for upgrade and includes such tasks as closing database connections, clearing pending workflow tasks, disabling customized triggers and editing and validating the storage control file.

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 Server Configuration Utility in upgrep mode. This utility performs a basic upgrade of the Siebel Database schema and loads repositories to prepare for the repository merge. Specifically, 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 Prior v.x.x Siebel 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
- **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.

- **Upgrade database (upgphys).** You run the Database Server Configuration Utility in upgphys mode. It further upgrades the Siebel Database with changes resulting from the repository merge and completes the database upgrade.

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

Specifically, this mode performs the following tasks:

- 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, and exports the logical schema definition to a file, schema.ddl

The customer repository definition file (custrep.dat) and logical schema definition file (schema.ddl) are used as input to the production upgrades.

- Renames the New Customer Repository to Siebel Repository
- Updates the schema version in S_APP_VER

Perform Upgrade Tasks (Production Test Environment)

This phase upgrades a production test environment Siebel Database to the new release allowing you to test how customizations work with the new release and to tune the upgrade scripts.

CAUTION: You are required to contact Siebel Expert Services for help in tuning your upgrade scripts. If you do not, you may invalidate your support agreement.

This phase includes the following tasks:

- **Prepare application data.** These tasks are about preparing application data for migration.
- **Prepare for Production Upgrade.** Run the Database Server Configuration Utility in Prepare for Production Upgrade mode. This mode examines the upgraded development environment database and generates SQL (dedup.jcl and dedup_prod.jcl files) that removes duplicate records from the intersection tables and sets up support for database aggregation.

You only have to run the utility in Prepare for Production Upgrade mode once—either in a production test environment or in the live production environment—to generate the dedup.jcl and dedup_prod.jcl files.

NOTE: You must define an ODBC connection to the development environment database before performing this upgrade step.

- **Upgrade database (upgrep + upgphys).** Run the Database Server Configuration Utility in upgrep + upgphys mode and enter configuration information for the production test environment. This step then:
 - Updates the configuration file (master.ucf) that will be used by the Siebel Upgrade Wizard

- Imports the repository and schema definition files from the development environment upgrade and uses these to generate the files used to perform a basic upgrade of the Siebel Database schema

Specifically, 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
 - Makes several administrative changes to table data, including updating the schema version in S_APP_VER.
- **Tune upgrade scripts (optional).** You can improve the performance of the production environment upgrade by tuning the production upgrade scripts in the test environment.

You should run several production upgrades against the test database. This allows you to understand the upgrade process before performing the production upgrade, to conduct performance testing, and to fine-tune the upgrade scripts. After carrying out thorough performance testing, you can perform the live production upgrade using the tuned upgrade files.

NOTE: If you have tuned the upgrade scripts, ensure that you copy these files to a safe location so that you can use them during the production upgrade.

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 involve preparing application data in the production database for migration.

- **Upgrade database (upgrep + upgphys).** You run the Database Server Configuration Utility in upgrep + upgphys mode and enter configuration information for the production environment. This includes defining an ODBC connection to the production environment Siebel database.

The Database Server Configuration Utility updates the upgrade configuration file with the production environment information you enter. When you run the Siebel Upgrade Wizard, it reads the production environment information from the updated configuration file and uses this information in generating the upgrade files used to upgrade the production environment Siebel database. This mode makes the same changes as when you ran the Database Server Configuration Utility in the production test environment.

NOTE: You do not have to run the utility in **Prepare for Production** mode before starting your production environment upgrade. You ran it as part of the production test environment upgrade. The required upgrade SQL commands have already been generated.

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 applications and include such things 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 application function followed by a full suite of regression and stress tests to verify the system is ready for production.

Related Topics

[“About the Database Server Configuration Utility” on page 37](#)

[“About the Siebel Upgrade Wizard and Driver Files” on page 40](#)

[“About the Repository Merge” on page 49](#)

[“About the Siebel Database Server” on page 52](#)

About Midtier-Centric and Mainframe-Centric Upgrades

Upgrades: All upgrades.

Environments: Development environment only.

When upgrading your development environment, you can choose whether to perform a mainframe-centric or midtier-centric upgrade.

All Siebel upgrades on DB2 for z/OS are initiated from a midtier platform (Windows or UNIX) and the DDL and DML files necessary for the upgrade are generated on this platform. In all production environment upgrades, the DDL and the data migration DML files are transferred to the DB2 host where they are unpacked and applied; this process is guided on the mainframe using REXX panels. This is a mainframe-centric upgrade.

Historically, all development environment upgrades have been midtier-centric, that is, only the DDL upgrade files are run directly on the mainframe; all other upgrade steps are performed from the midtier and applied to the z/OS host using a Windows- or UNIX-based DB2 client. Midtier centric development upgrades are very similar to non-z/OS development upgrades.

Since the development environment upgrade was enhanced in Version 7.7.2, however, all development upgrade DDL (CREATE, DROP, ALTER, and GRANT) and DML (INSERT, UPDATE, DELETE, and SELECT) can be executed on the mainframe host. Based on your business requirements, you can choose to perform either a midtier-centric or mainframe-centric development environment upgrade.

NOTE: By default, the Siebel software contains the midtier-centric development environment upgrade files. If you want to perform a mainframe-centric development environment upgrade, you need to copy and replace these files to generate the appropriate upgrade output files. See [“Generating Development Environment Mainframe-Centric Upgrade Files”](#) on page 202 for further information.

Source and Target Databases

When performing a development environment upgrade from the midtier, the source database is upgraded in place. For recovery purposes, in-place upgrades are not supported for mainframe-centric development or production upgrades. Instead, a target database is built for the new release and data is unloaded from the existing source database and migrated in the appropriate format to the upgraded database. The source database is not upgraded, although minor modifications are made to it during the upgrade process. The source and target databases can be in different DB2 subsystems or in the same subsystem.

Upgrading a development environment from the midtier is a simpler, more automated process than performing a mainframe-centric development upgrade. However there are advantages and disadvantages to each type of upgrade, as shown in [Table 8](#).

Table 8. Midtier-Centric and Mainframe-Centric Development Environment Upgrades

Upgrade	Advantage	Disadvantage
Midtier-centric development environment.	<ul style="list-style-type: none"> ■ This type of upgrade follows previous Siebel upgrade flows and allows you to work with existing expertise. ■ Requires less manual intervention and is a less complex procedure. 	<ul style="list-style-type: none"> ■ Does not allow you to take advantage of enhancements built into the mainframe-centric development environment upgrade. ■ This upgrade is not easy to restart if failures occur so you must back up your database before upgrading. When running a job, your options are to: <ul style="list-style-type: none"> ■ Stop on every error. Some SQL statements return acceptable errors so, if you select this option, you may have to manually remove statements already processed and rerun the job. ■ Process complete file. If the complete file is processed and an unacceptable error is returned, it may be necessary to restore the database from backup and restart the upgrade.
Mainframe-centric development environment.	<ul style="list-style-type: none"> ■ You can gain experience performing this type of upgrade that you can apply during your production environment upgrade. ■ You submit SQL files in a JCL environment. If a job fails, you know immediately and can correct the problem and re-submit the job before data is corrupted. ■ This process can be audited. ■ Involves the use of familiar, native z/OS tools and utilities. 	<ul style="list-style-type: none"> ■ You require a TSO account and the correct access permissions and authority levels. ■ Extra disk space is required because a source and target database are used in the upgrade. ■ You must acquire the mainframe-centric development upgrade files and manually replace the related midtier-centric files.

About the Database Server Configuration Utility

Upgrades: All upgrades.

Environments: All environments.

The Database Server Configuration Utility performs the upgrep and upgphys portions of an environment upgrade. You must also run the Database Server Configuration Utility in Prepare for Production Upgrade mode before the production environment upgrade.

The Database Server Configuration Utility requests information from you about the upgrade environment and creates an upgrade configuration file. It then calls a driver that uses the environment information to create the SQL scripts required to upgrade your database.

After you run the Database Server Configuration Utility, you run the Siebel Upgrade Wizard. The Siebel Upgrade Wizard opens a driver file containing the steps for the upgrade and executes these steps.

To upgrade a development environment, production test environment, or production environment, you must run the Database Server Configuration Utility (and Siebel Upgrade Wizard) several times, as shown in [Table 9](#).

Table 9. Database Server Configuration Utility Modes

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

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

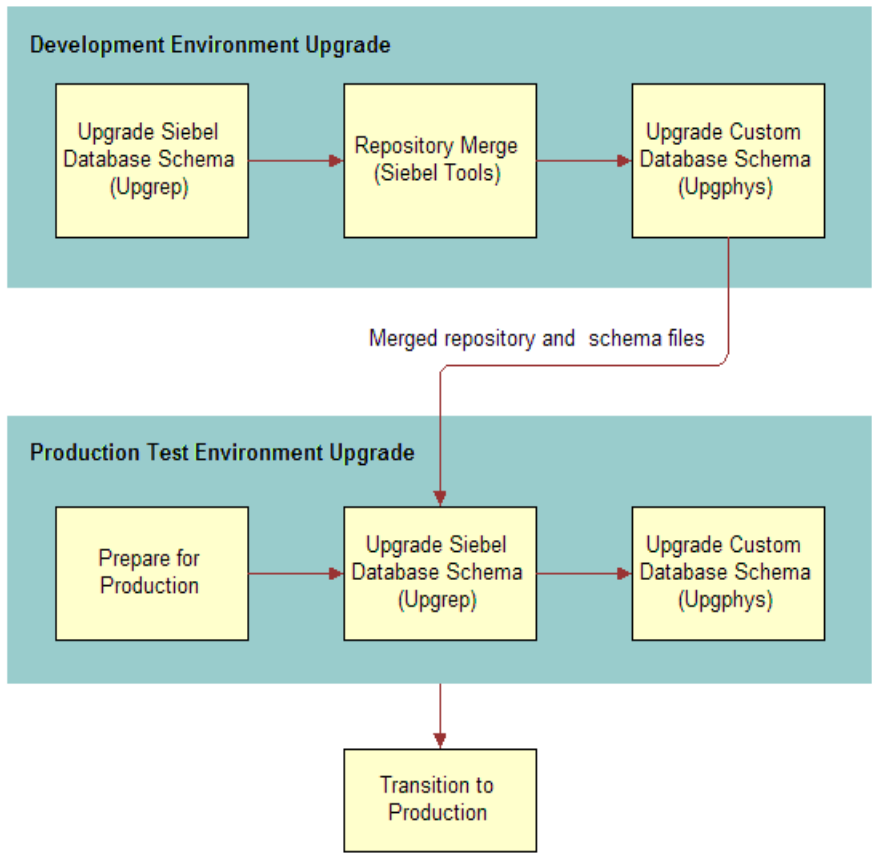


Figure 2. Database Server Configuration Utility and Siebel Tools Repository Merge

How the Upgrade Configuration File and SQL Files Are Created

When you run the Database Server Configuration Utility, it does the following:

- Collects configuration information
- Creates a master upgrade configuration file (UCF). This file maps the information you entered in the Database Server Configuration Utility to environment variables. When the Siebel Upgrade Wizard is performing the steps in a driver file, it uses these variables to execute the command contained in each step.
- Forwards 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.
- Prompts you to start the Siebel Upgrade Wizard. The wizard uses the UCF file to execute the SQL files and other input files against the Siebel Database.

In some cases, you will need to modify the generated SQL files as required by Siebel Release Notes, Technical Notes, or Siebel Alerts before you run the Siebel Upgrade Wizard. To do this, answer No when prompted to run the Siebel Upgrade Wizard. Then, edit the SQL files and manually launch the Siebel Upgrade Wizard. See [“Starting the Siebel Upgrade Wizard” on page 170](#) for more information.

How To Locate Master Configuration Files

Master upgrade configuration files are stored in the following location:

Windows: `SI EBEL_ROOT\bin`

UNIX: `$SI EBEL_ROOT/bin`

Master upgrade configuration files use the following naming convention:

`master_UPGRADEOPTION_ENVIRONMENT_VERSION.ucf`

where:

- `UPGRADEOPTION` = the upgrade process you are performing
 - Siebel Database schema upgrade = `upgrep`
 - Custom database schema upgrade = `upgphys`
 - Prepare for production upgrade = `prepare_for_production_upgrade`
- `ENVIRONMENT` = the environment that you are upgrading
 - Development environment upgrades = `dev`
 - Production environment upgrades = `prod`
- `VERSION` = the version from which you are upgrading

For example, if you are upgrading from Siebel 7.0.4, the UCF file generated from the development environment `upgrep` is as follows:

`master_upgrep_dev_704.ucf`

The UCF file generated from the Prepare for Production Upgrade mode is as follows:

`master_prepare_for_production_upgrade.ucf`

About Parameter Validation

The Database Server Configuration utility validates certain parameters, and does not proceed if you enter an invalid parameter. The configuration wizard validates the following information:

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

- That the directories chosen exist

About the Siebel Upgrade Wizard and Driver Files

Upgrades: All upgrades.

Environments: All environments.

The Siebel Upgrade Wizard makes changes to the Siebel Database using the SQL files and the upgrade configuration file (UCF) generated by the Database Server Configuration Utility. After the Siebel Upgrade Wizard starts, it executes this process:

- Reads the UCF
- Calls a driver
- Passes the information in the UCF to the driver, which then passes UCF file information to:
 - the ddlimp utility, which executes ddl-type SQL commands
 - the dataimp utility, which executes data-related SQL commands

The Siebel Upgrade Wizard also does the following:

- Upgrades Siebel seed data
- Loads the following repositories for development environment upgrades:
 - Prior Standard Repository
 - New Standard Repository
 - New Customer Repository

If ddlimp or dataimp encounter errors, they exit. When the errors have been corrected, you can launch the Siebel Upgrade Wizard, and the upgrade resumes from where it stopped.

Upgrade Wizard Pauses

The Siebel Upgrade Wizard pauses on three occasions during the upgrade process. At each pause, you must transfer the files that the Upgrade Wizard has generated to the z/OS host and execute them before you continue the upgrade. The pauses occur as follows:

- 1 Launch the Wizard to begin the upgrade and generate PRET files (to Pause #1).
- 2 Resume the Upgrade Wizard to generate the Data Migration files (to Pause #2).
- 3 Resume the Upgrade Wizard to generate the Gen Primary and new index files (to Pause #3).
- 4 Resume the Upgrade Wizard to complete the upgrade process.

The Upgrade Wizard generates separate DDL and DML files. The files generated by the Upgrade Wizard are output by default to the dbsrvr\db2390\dboutput\upgrade directory or to the DDL Output Directory you specified when you ran the Database Server Configuration Utility.

Driver Files

The Siebel Upgrade Wizard performs the upgrade by executing the commands and SQL scripts contained in driver files. Driver files are in ASCII text format and are organized into steps. Each step specifies which command or SQL script to run.

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 Siebel Upgrade Wizard.

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

- driver_upgrep_dev_621_mf.ucf
- driver_upgphys_dev_77_mf.ucf

Here is an excerpt from a driver file that controls a development environment upgrep from Release 7.7 SIA to release 7.8 SIA. The excerpt contains three steps:

[Sql File Entry 21]

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 22]

Type = FileExecute

File Name = \$SiebelRoot\bin\odbcsql

Check Return Code = 1

Return Code Compliance = 0

16 Bit App = 0

Command Line = /s "\$ODBCDataSource" /u \$UserName /p \$Password /q \$DatabaseOwner /separator / /a /c rem /I \$SiebelLogDir/del appkey.log \$DbsrvrRoot/\$DatabasePlatform/del appkey.sql /v

Number of 10 Second Wait Loops = 2000

Return Code = 0

Title = Delete old license key

Title Message Num = 0

Estimated Disk Space = 0

Backup Db = 0

Parallelizable Item = 0

Prompt User For Status = 0

[File Execute Entry 23]

Type = FileExecute

File Name = \$SiebelRoot\bin\dataimp

Check Return Code = 1

Return Code Compliance = 0

16 Bit App = 0

Command Line = /u \$UserName /p \$Password /c "\$ODBCDataSource" /d \$DatabaseOwner /f \$SeedFileName /h Log /w y /q 100 /e n /i \$DbsrvrRoot/\$Language/\$DatabasePlatform/seedupg0.i np /l \$Siebel LogDir/seedupg0. log /A 1 /B y

Number of 10 Second Wait Loops = 2000

Return Code = 0

Title = Import Siebel seed data

Title Message Num = 0

Estimated Disk Space = 0

Backup Db = 0

Parallelizable Item = 1

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\DB2390\UPGRADE\VERSION*

UNIX: *DBSRVR_ROOT/DB2390/UPGRADE/VERSION*

where:

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:

driver_upgrep_dev_77.ucf

Related Topics

[“About the Database Server Configuration Utility” on page 37](#)

[“About the Repository Merge” on page 49](#)

Job Flow of a Production Database (Mainframe-Centric) Upgrade

Upgrades: All upgrades.

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

This topic describes the major steps in a production database upgrade using the Database Server Configuration Utility and Siebel Upgrade Wizard. This topic does not describe all the pre- and post-upgrade tasks you must complete for a production environment. For a description of all the steps in upgrading a production environment, see [“Process of Upgrading a Production Environment” on page 77](#).

The production *environment* upgrade job flow differs from a development environment upgrade job flow primarily in that a repository merge is not required. The repository (custrep.dat) and the logical schema (schema.ddl) are exported from the upgraded development environment and used in the production upgrade.

A production *database* upgrade job flow, as illustrated in Figure 3, is largely the same as the job flow of the *upgrep* stage of a mainframe-centric, development database upgrade.

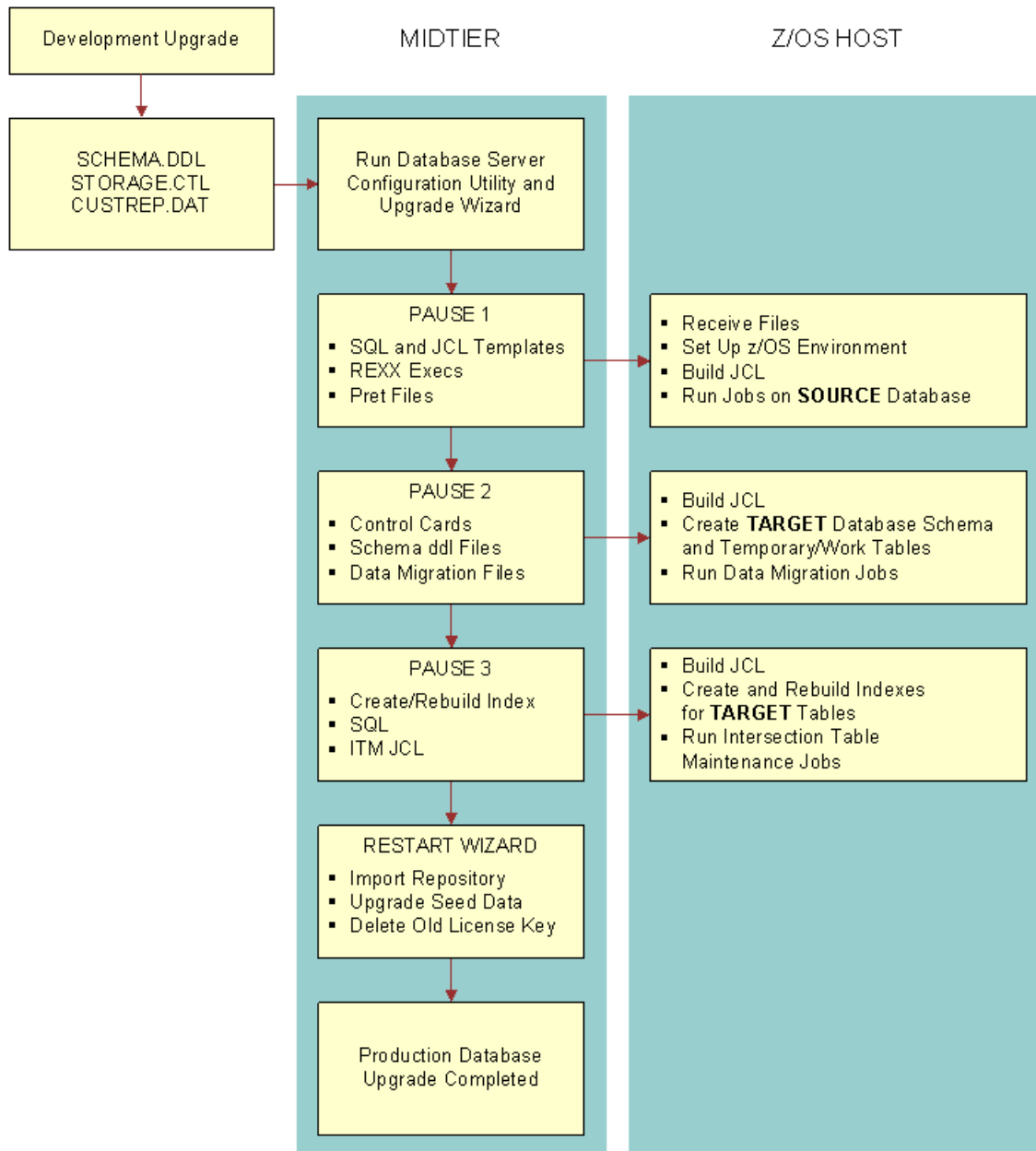


Figure 3. Flow of a Production Database Upgrade

The main steps in a production upgrade illustrated in Figure 3 are as follows:

1 Inputs to the Production Upgrade.

- ❑ Load target tables.
- ❑ Build indexes on the source table.
- ❑ Rebuild the source indexes.
- ❑ Migrate PRESCHM, product configurator, and ISS data in the target schema.

There are optional data migration scripts for eChannel and Household data and for Siebel Financial Services (FINS) applications.

- 6 Pause # 3.** Restart the Upgrade Wizard on the midtier to generate the index rebuild and the gen_primary index jobs. The Upgrade Wizard stops at the third pause.
 - a** FTP the files generated on the midtier by the Upgrade Wizard to the host.
 - b** Build the JCL for the index create and rebuild jobs.
 - c** Load the target log tables with index rebuild information.
 - d** Drop non-unique indexes that are in the old schema format
 - e** Create non-unique indexes in the new schema format
 - f** Rebuild the indexes.
 - g** Run the gen primary jobs in parallel to update data in the target tables.
 - h** Create indexes in the new schema format for EIM tables.
 - i** Build and run RUNSTATS JCL for the DB2 objects for which statistics have not been collected during the upgrade process.
 - j** Run the Intersection Table Maintenance jobs to identify and resolve potential duplicate row IDs from intersection tables.
- 7 Complete Upgrade Processing.** Restart the Upgrade Wizard on the midtier to complete the upgrade processing. The Upgrade Wizard automatically runs the remaining DML upgrade jobs from the midtier to perform a number of tasks, including the following:
 - a** Deletes the old license key.
 - b** Verifies the repository name and imports the New Customer Repository from the upgraded development environment.
 - c** Upgrades seed data to the new version.

The production database upgrade is now completed.

Key Members in a Mainframe-Centric Upgrade

Members in the *DSNHLQ.SIEBEL.EXEC* contain useful information that you can refer to when performing your mainframe-centric upgrade. These key members are listed in [Table 10](#).

Table 10. Key Members in the *DSNHLQ.SIEBEL.EXEC*

Member Name	Describes
@JOBPRFX	Shows details of the job prefixes chosen for the upgrade.
@LASTJOB	Lists the job name of the last job for each step of the upgrade, for example: <ul style="list-style-type: none"> ■ PRESCHM: S003562Z ■ PRESCHF: S006463Z
@SRCTAR	Lists details of the source and target environments, for example: <ul style="list-style-type: none"> ■ SOURCE: HOSTNAME=QA02 DB2SYS=Q2F1 TABLEOWNER=CQ2F1001 ■ TARGET: HOSTNAME=QA02 DB2SYS=Q2F1 TABLEOWNER=CQ2F1900
@TBOSRC	Provides source tableowner details.
@TBOTAR	Provides target tableowner details.
@UPGPATH	Lists the upgrade path and the upgrade status, that is, the last step executed, for example: PATHID=2 STEP=4 - NEW SIEBEL GP INDEX REBUILD JOBS GENERATED
@UPGTYPE	Shows the type of upgrade being run, for example, SIA753.

About JCL Job Generation

- The *DSNHLQ.SIEBEL.EXEC* data set contains base job cards for all the upgrade jobs. The upgrade process uses these job cards as a base for all jobs of a particular type and then generates an individual job card for each job based on the information in the base card and in the @JOBPREFIX member of the *DSNHLQ.SIEBEL.EXEC* data set.

For example, the base job card for PRESCHM jobs is JCPRES. An individual job card is created for each PRESCHM job based on information in both of the following:

- *DSNHLQ.SIEBEL.EXEC*(@JOBPREFIX) – job prefix
- *DSNHLQ.SIEBEL.EXEC*(JCPRES) – base job card

Once job cards have been generated, changing the information in the @JOBPREFIX member does not cause job card details to change.

- Load and Unload job procedures are located in the *DSNHLQ.SIEBEL.PROC* data set.

- Some JCL jobs, for example, some Data Migration jobs, have dependencies on other jobs. These jobs are submitted in a specific order, with subsequent jobs only being submitted when the current job has completed successfully. The @TRIGGER member in the relevant JCL data set shows the order in which the jobs are to be submitted, for example, see *DSNHLQ.SIEBEL.PRESCHM.JCL(@TRIGGER)* for details of the submission sequence for PRESCHM jobs.

About the Override File

Upgrades: Releases 6.x, 7.0.x, 7.5.x.

Environments: Development environment only.

The override.inp file allows you to override your existing database storage layout when you upgrade to Release 7.8.

Since Release 7.7, the Siebel database schema structure has the following characteristics:

- One table in each tablespace
- One tablespace in each database

This layout is referred to as the *1:1:1* model. See [“Planning Changes to the Physical Layout of the Schema” on page 88](#) for further information.

If you are upgrading from a pre 7.7 release of Siebel applications, you may want to switch to the *1:1:1* model from your existing database schema structure because it is more efficient. There are three options available when upgrading to the current release:

- Preserve the existing model of multiple tables per tablespace and only build new tables with the *1:1:1* model.

To use this option, extract a storage control file and merge it with an existing template. All source definitions are preserved and new tables are built according to the definitions supplied in the template storage control file.

- Preserve the existing model for most of the source tables but specify specific source tables to use the *1:1:1* model by adding the table names to the override.inp file. By default, the override.inp file has two table entries:

- S_DOCK_TXN_LOG
- S_SERVICE_SCRIPT

The existing definition for these two tables is *not* preserved from the source to the target. Therefore, these two tables can be built using the new model as specified in the template.

- Move all source tables from the current model to the *1:1:1* model.

To use this option, do *not* use the extract and merge process to create the storage control file. Instead, use the preconfigured storage control file. In the preconfigured storage control file, all tables are created with the *1:1:1* paradigm. Therefore, each table is created in its own database or tablespace and the data is moved from the source to the target table.

The use of the preconfigured storage control file allows you to use the *1:1:1* model for all tables other than obsolete tables and customer extended tables. The preconfigured storage control file does not apply to obsolete or customer extended tables so there are no entries for them. You cannot put these into the `override.inp` file since there is nothing to override and the tables are not in the storage control file. In this case, the existing schema is preserved.

The tablespaces are not cleaned up unless they are empty. Some of the obsolete and customer extended tables are still in those tablespaces so the upgrade process cannot just do a general cleanup.

About the Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

The Siebel Repository consists of records stored in a group of tables in the Siebel Database. There are 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.

Think of the Siebel Repository as these two types of records, rather than the tables that store the records. Repository records include a repository ID that lists the ID of the repository to which each record belongs. The repository ID forms part of the user index for these records and allows several repositories to be stored in the same set of tables.

List-Applets in Siebel Tools display the information in the repository records.

Upgrading the Siebel Database adds new records to the tables that store the repository records. These new records make up three repositories, as shown in [Table 11](#).

Table 11. Development Database Repositories

Repository Name	Siebel Tools Name	Description	Added During Upgrade?
Prior V7.x Siebel Repository	Prior Standard	The standard Siebel Repository for your installed release (the one you are upgrading from).	Yes
Prior Customer Repository	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.	No
New Siebel Repository	New Standard	The Siebel Repository for the Siebel release to which you are upgrading. This repository defines the new Siebel release.	Yes
New Customer Repository	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.	Yes

How the Merge Works When You Are Upgrading from Siebel 7.x

The repository merge does the following things:

- Identifies customizations you have made in the Prior Customer Repository and uses decision criteria to merge them into the New Customer Repository
- Identifies logical schema changes you have made in the Prior Customer Repository and tries to include them in the logical schema in the New Customer Repository
- Upgrades both the base repository and all locales
- Writes a log describing actions, conflicts, and errors
- After the merge is complete, launches postmerge utilities that perform further upgrades to layout objects such as Web templates and controls

The merge process makes numerous record comparisons and inserts. As a result, the merge is resource-intensive and typically takes twelve to eighteen hours to complete. The time required varies depending on the complexity of customizations and amount of computing power.

How the Merge Works When You Are Upgrading from Siebel 6.2.1 FINS

The repository merge does the following things:

- Identifies customizations you have made in the Prior Customer Repository and uses decision criteria to merge them into the New Customer Repository
- Identifies logical schema changes you have made in the Prior Customer Repository and tries to include them in the logical schema in the New Customer Repository
- Upgrades both the base repository and all locales
- Writes a log describing actions, conflicts, and errors
- After the merge is complete, launches postmerge utilities that perform further upgrades to layout objects such as Web templates and controls

The merge process makes numerous record comparisons and inserts. As a result, the merge is resource-intensive and typically takes five to seven hours to complete. The time required varies depending on the complexity of customizations and amount of computing power.

How the Merge Handles Your Customizations

The repository merge handles customizations in the Prior Customer Repository as follows:

- If an object has not changed in the New Siebel Repository but was customized in the Prior Customer Repository, the customization is added to the object in the New Customer Repository.
- If a new object was created in the Prior Customer Repository, it is added to the New Customer Repository.
- If an object is present in the Prior Standard Repository and Prior Customer Repository, but is not present in the New Siebel Repository, the object is obsolete and is not used in the New Customer Repository.
- If an object attribute has changed in the New Siebel Repository and also has changed in the Prior Customer Repository, a conflict occurs. An alternative definition for conflicts is that the attribute value is different in all three repositories: Prior Standard, Prior Customer, and New Siebel.

The merge resolves about 90% of conflicts by using the object attribute as it is defined in the New Siebel Repository.

- If an object was deleted from the Prior Customer Repository, it is added to the New Customer Repository. The following objects are exceptions. If they were deleted from the Prior Customer Repository, they are not added to the New Customer Repository:
 - Control
 - List Column
 - Page Tab
 - Chart
 - Applet Web Template Item
 - View Web Template Item

About the Siebel Database Server

Upgrades: All upgrades.

Environments: All environments.

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 in the sense that it participates 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:\sea7x\si_ebsrvr` (Windows), then the Siebel Database Server will be installed in `C:\sea7x\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 `$IEBEL_ROOT` (Windows), `$SIEBEL_ROOT` (UNIX).

4

How to Perform the Upgrade

This chapter provides a roadmap for performing each type of upgrade to Siebel 7.8. Each roadmap lists the processes and tasks you must follow to complete the upgrade. Print the relevant roadmap and use it to guide you in carrying out the upgrade.

This chapter includes the following topics:

- [“Roadmap for Performing the Upgrade”](#)
- [“Process of Planning the Upgrade” on page 55](#)
- [“Process of Upgrading a Development Environment from the Midtier” on page 55](#)
- [“Process of Upgrading a Development Environment from the Mainframe” on page 62](#)
- [“Process of Upgrading a Production Test Environment” on page 70](#)
- [“Process of Tuning Upgrade Performance” on page 75](#)
- [“Process of Upgrading a Production Environment” on page 77](#)

Related Topics

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

[Chapter 3, “How the Upgrade Works”](#)

Roadmap for Performing the Upgrade

Upgrades: All upgrades.

Environments: All environments.

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

Each roadmap 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 roadmap. When you have completed all the processes in the roadmap, the upgrade is complete.

Depending on your 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.

- [“Roadmap for Upgrading from Siebel Financial Services 6.2.1 and Siebel 7.x” on page 54](#)
- [“Roadmap for Upgrading Without a Development Environment” on page 54](#)

Roadmap for Upgrading from Siebel Financial Services 6.2.1 and Siebel 7.x

If you are upgrading from Release 6.2.1 of Siebel Financial Services or Release 7.x, complete the processes in this roadmap in the order shown:

- 1 [“Process of Planning the Upgrade” on page 55](#)
- 2 [“Process of Upgrading a Development Environment from the Midtier” on page 55](#) or [“Process of Upgrading a Development Environment from the Mainframe” on page 62](#)
- 3 [“Process of Upgrading a Production Test Environment” on page 70](#)
- 4 [“Process of Tuning Upgrade Performance” on page 75](#)
- 5 [“Process of Upgrading a Production Environment” on page 77](#)

For a description of the differences between production and development upgrades, see [“About Upgrade Environments” on page 26](#).

Depending on your business requirements, you can upgrade your development environment from the midtier (midtier-centric approach) or mainframe (mainframe-centric approach). For more information, see [“About Midtier-Centric and Mainframe-Centric Upgrades” on page 34](#).

Roadmap for Upgrading Without a Development Environment

If you are upgrading and do not have a development environment, complete the processes in this roadmap 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 the Upgrade” on page 55](#).
- 2 [“Preparing for a Non-Development-Environment Upgrade” on page 180](#).
- 3 Perform a production test upgrade. See [“Process of Upgrading a Production Test Environment” on page 70](#).
- 4 Upgrade your production environment. See [“Process of Upgrading a Production Environment” on page 77](#).

Process of Planning the Upgrade

Upgrades: All upgrades.

Environments: All environments.

This process is part of an upgrade roadmap. See the roadmaps earlier in this chapter.

To plan the upgrade, read the following:

- 1 [Chapter 3, “How the Upgrade Works.”](#)
- 2 [Chapter 5, “Database and UI Upgrade Planning.”](#)
- 3 [Chapter 6, “Application Upgrade Planning.”](#)
- 4 [Appendix A, “Tables Modified or Seeded During Upgrade.”](#)
- 5 [Appendix B, “Siebel Marketing Upgrade Reference.”](#)
- 6 Review upgrade-related publications on Oracle Technology Network or My Oracle Support. See [“Important Upgrade Planning Resources”](#) on page 83.

Process of Upgrading a Development Environment from the Midtier

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This process is part of an upgrade roadmap. See [“Roadmap for Performing the Upgrade”](#) on page 53.

This topic lists the steps required to upgrade a development environment to the current release from the midtier. 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 Release 6.2.1 of Siebel Financial Services (FINS) are marked as such.

Upgrade Third-Party Software

- 1 Upgrade third-party software as required due to dependencies on 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.

For further information, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are documented in *Siebel Maintenance Release Guide* on My Oracle Support.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using and *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

- 2 Install the Siebel Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).

The upgraded Siebel Servers will not work correctly with the RDBMS server until after you have upgraded the Siebel Database to the new release.

- 3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade.
- 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*.

NOTE: You do not install a new Siebel database as part of upgrading the Siebel Enterprise.

Upgrade the RDBMS

- 6 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade. For information on supported RDBMS systems, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Identify and Resolve Duplicate Row IDs

- 7 **6.2.1 upgrades only:** [“Identifying and Resolving Duplicate Row IDs” on page 145.](#)

After you install the Siebel Database Server software, but before you upgrade to this release, you must identify and resolve any duplicate row IDs in your Siebel Database.

Preupgrade Tasks for the Siebel Database

- 8 Review Siebel Technical Notes and Siebel Alerts on upgrading to the current release. Make revisions to the upgrade process as required. See [“Important Upgrade Planning Resources” on page 83.](#)
- 9 Review 477519.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 521. It contains important changes to database records and repository objects that must be made during the upgrade. See [“Important Upgrade Planning Resources” on page 83.](#)
- 10 Review guidelines for configuring the RDBMS. See *Implementing Siebel Business Applications on DB2 UDB for z/OS*.
- 11 Verify all developers have checked in their projects.

- 12 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 13 Stop the Siebel Server and the Siebel Gateway Name Server.
- 14 Verify there are no open database connections.
- 15 Perform the tasks in [Chapter 7, “Basic Database Preparations.”](#)

Preupgrade Tasks for a Development Environment Upgrade

- 16 Perform the tasks in [Chapter 9, “Preparing a Development Environment for Upgrade.”](#)

Preupgrade Tasks for Application Data

- 17 Perform the relevant tasks in [Chapter 8, “Preparing Application Data for Upgrade.”](#)

Some of these tasks are optional, depending on the currently installed Siebel products and your upgrade path. Review and perform these tasks as necessary.

Upgrade Siebel Database Schema (upgrep)

- 18 Install Siebel Tools on development workstations. Keep at least one copy of the previous version of Siebel Tools. You will need it to perform repository operations before the repository merge.
- 19 [“Migrating Repository Objects to the Standard UI” on page 271.](#)
- 20 Back up the development database. (If you backed up the database as part of an RDBMS upgrade, ignore this step.)
- 21 Run the Database Server Configuration utility on the midtier and enter development environment information to create the upgrade configuration file:
 - [“Preparing to Run the Database Server Configuration Utility” on page 164.](#)
 - [“Running the Database Server Configuration Utility Under Windows” on page 168.](#)
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169.](#)

When running the Database Server Configuration utility, choose the following settings:

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

The master UCF file is updated with the development environment configuration.

- 22 Enter Yes to launch the Siebel Upgrade Wizard to start the database upgrade process. SQL commands are executed on the development environment database until the Upgrade Wizard stops at the First Pause.
- 23 Review the following topic and perform the procedure if applicable: [“Migrating Address Data from Custom Extension Columns” on page 173.](#)
- 24 [“Executing SQL Files After the First Pause” on page 183.](#)
- 25 [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 185.](#)

- 26 [“Executing SQL Output Files After the Second Pause - Stage 1 of 2”](#) on page 187.
- 27 [“Deploying Stored Procedures and User-Defined Functions”](#) on page 188.
- 28 [“Executing SQL Output Files After the Second Pause - Stage 2 of 2”](#) on page 193.
- 29 [“Resuming the Siebel Upgrade Wizard After the Second Pause”](#) on page 193.
- 30 [“Executing SQL Output Files After the Third Pause”](#) on page 195.
- 31 [“Resuming the Siebel Upgrade Wizard After the Third Pause”](#) on page 195.
- 32 [“Reviewing Upgrade Log Files for Errors”](#) on page 256.
- 33 If the upgrade contains unacceptable errors, do the following:
 - Restore the backup of the database.
 - Correct the errors.
 - Rerun the Database Server Configuration utility.
- 34 [“Manually Archiving Upgrade Log Files”](#) on page 259.
- 35 [“Installing New License Keys After Upgrade”](#) on page 180.
- 36 **Multilingual deployments.** If you have multilingual deployments, perform the steps in 477094.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 447. This document describes how to import language-specific repository strings and seed data into the upgrade repositories.
- 37 Back up the upgraded database repository after a successful upgrade of the Siebel Database Schema.

Prepare for Repository Merge

- 38 Set the Upgrade Ancestor property for copied objects. See [“Configuring Objects to Inherit Upgrade Behavior”](#) on page 270.
- 39 [“Migrating Repository Objects to the Standard UI”](#) on page 271.
- 40 [“About Backing Up the New Customer Repository or Database Schema”](#) on page 275.
- 41 **6.2.1 upgrades only:** [“Running the Repository Preparation Wizard on Release 6.2.1 Repository”](#) on page 277.
- 42 Execute the REORG utility on tables that receive a large number of inserts during the repository import process. See [“About Reorganizing Tables Before the Repository Merge”](#) on page 275 for further information.
- 43 Run database statistics on the Siebel Database. For further information on running database statistics, see [“Updating Statistics”](#) on page 197. Running statistics on the Siebel Database improves merge performance.

If upgrading from Release 7.7, run statistics specifically on the S_SYM_STR and SYM_STR_INT tables. If you are upgrading from a pre-7.7 release, the S_SYM_STR and SYM_STR_INT tables are not populated until the merge is completed so you do not need to run statistics on them at this point.

Perform Repository Merge

CAUTION: The Repository merge process cannot be stopped and restarted so ensure you have backed up the database schema or the New Customer Repository before starting the merge.

- 44 [“Performing a Repository Merge”](#) on page 279.
- 45 [“Determining If a Repository Merge Was Successful”](#) on page 288.
- 46 If the repository merge contains unacceptable errors, do the following:
 - a Restore the backup of the database or New Customer Repository.
 - b Correct the errors.
 - c Rerun the Database Server Configuration utility.
- 47 [“Migrating Custom Workflows”](#) on page 289.
- 48 Back up the database.

Migrate 6.2.1 Customizations

6.2.1 upgrades only: The steps in [“Migrate 6.2.1 Customizations”](#) apply only to upgrades from Release 6.2.1.

- 49 [“Identifying What Will Be Migrated”](#) on page 309.
- 50 [“Exposing Hidden Properties of Applets and Views”](#) on page 310.
- 51 [“Running the Siebel Web Client Migration Wizard”](#) on page 311.
- 52 [“Reviewing Applet and View Migration”](#) on page 314.
- 53 [“Running the Web Layout Wizard”](#) on page 315.
- 54 [“Migrating Scripts Attached to Controls”](#) on page 316.
- 55 [“Migrating Scripts Attached to Applets”](#) on page 318.
- 56 [“Migrating Business Component, Business Service, and Application Scripts”](#) on page 321.
- 57 [“Migrating Outbound COM Interfaces”](#) on page 322.
- 58 [“Migrating Inbound COM Interfaces”](#) on page 323.
- 59 [“Regenerating the Postmerge Utilities Log”](#) on page 326.
- 60 [“Reviewing Customized Business Components”](#) on page 326.

Perform Postmerge Tasks

- 61 [“Running the Postmerge Utilities”](#) on page 329.
- 62 (Optional.) Compile an SRF file to help review the UI. See [“Creating a New SRF File”](#) on page 297.
- 63 [“Reviewing Attribute Conflicts in the Repository Merge”](#) on page 291.
- 64 Perform the tasks in Chapter 17, [“Postmerge Development Tasks.”](#)
- 65 [“Generating EIM Temporary Columns After a Repository Merge”](#) on page 294.

Upgrade Physical Custom Database Schema (upgphys)

You must now upgrade the physical custom database schema.

66 Run the Database Server Configuration utility on the midtier and enter appropriate values to configure the Database Server to run the physical schema upgrade:

- [“Preparing to Run the Database Server Configuration Utility” on page 164.](#)
- [“Running the Database Server Configuration Utility Under Windows” on page 168.](#)
- [“Running the Database Server Configuration Utility Under UNIX” on page 169.](#)

When running the Database Server Configuration utility, choose the following settings:

- **Upgrade Options:** Upgrade Siebel Database Schema (upgphys)
- **Environment Type:** Development

67 Enter Yes to launch the Siebel Upgrade Wizard. SQL commands are executed on the development environment database until the Upgrade Wizard stops at the First Pause.

68 Run the files generated by the Upgrade Wizard as described in [“Synchronizing the Schema” on page 198.](#)

69 [“Dedup Files” on page 199.](#)

70 [“Migrating Custom Business Component Configurations” on page 199.](#)

71 [“Reviewing Upgrade Log Files for Errors” on page 256.](#)

72 If the upgrade contains unacceptable errors, do the following:

- a** Restore the backup of the database.
- b** Correct the errors.
- c** Rerun the Database Server Configuration utility.

73 [“Manually Archiving Upgrade Log Files” on page 259.](#)

74 Back up the upgraded database.

75 [“Deleting Redundant Upgrade Files” on page 181.](#)

76 [“Creating a New SRF File” on page 297.](#)

Postupgrade Tasks for Environment Setup

77 [“Updating File System Attachments” on page 355.](#)

78 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide* and [“Regenerating the Database Template File” on page 348.](#)

79 Extract developers' databases. See *Siebel Remote and Replication Manager Administration Guide* and [“Extracting Developers or Siebel Tools Clients” on page 349.](#)

80 Initialize the local database on development machines.

81 [“Checking for Inactivated EIM Table Columns” on page 353.](#)

- 82 Reset upgrade-specific parameters back to their defaults. See [“Resetting Database Server Configuration Parameters” on page 349](#).
- 83 Run database statistics. For more information, see [“Migrating Custom Business Component Configurations” on page 199](#).

NOTE: The development environment is now upgraded. The remaining sections deal with configuration and validation tasks.

Postupgrade Tasks for Configuration

- 84 [“Reviewing Deleted Objects in the Repository Merge” on page 292](#).
- 85 [“Reviewing Obsolete Objects in the Repository Merge” on page 293](#).
- 86 Resolve any business component and join conflicts.
- 87 [“Reapplying Custom Columns to the Siebel Database” on page 345](#).
- 88 [“Deleting Duplicate EIM Mappings” on page 347](#).
- 89 [“Deleting Unneeded Repository Files” on page 297](#).
- 90 If you exported data from interface tables before the upgrade, review the database and import the data as desired.
- 91 [“Upgrading to RC2 or AES Encryption” on page 96](#).

Configure for Globalization

- 92 Install language packs for new languages. See the *Siebel Installation Guide* for the operating system you are using.
- 93 [“Upgrading to the Symbolic String Model” on page 349](#).
- 94 [“Setting Up Your Environment to Support Global Time Zone” on page 362](#).

Deploy Workflows

Activate and deploy workflows. To perform these tasks, see *Siebel Business Process Designer Administration Guide*.

- 95 [“Upgrading Seeded Workflows” on page 388](#).
- 96 [“Upgrading Inbound Workflows” on page 388](#).

Verify Application Integration

- 97 Verify that EAI and EIM integrations are set up correctly. For information on using EAI and EIM, see *Overview: Siebel Enterprise Application Integration* and *Siebel Enterprise Integration Manager Administration Guide*.
- 98 [“Updating Enterprise Application Integration \(EAI\) After Upgrade” on page 363](#).

Postupgrade Tasks for Application Administration

- 99 Review the results of the Person and Organization merge. Make configuration changes as required.
- 100 [“Generating Reporting Relationships After Upgrade” on page 360.](#)

Perform System Tests

- 101 Perform the tasks in [Chapter 21, “Postupgrade Tasks for Applications”](#) as needed to upgrade test data in the development environment.
- 102 Use available test data to perform unit testing. Validate application function in the following areas:
 - User interface
 - Data interfaces
 - Integrity of migrated data
 - Workflow function
- 103 If you revise repository objects or schema definitions, regenerate the schema.dll and custrep.dat files. See [“Regenerating the Repository Definition Files” on page 295.](#)

Process of Upgrading a Development Environment from the Mainframe

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment.

This process is part of an upgrade roadmap. See the roadmaps earlier in this chapter.

This topic lists the steps required to upgrade a development environment from the mainframe 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 Release 6.2.1 are marked as such.

Upgrade Third-Party Software

- 1 Upgrade third-party software as required due to dependencies on 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.

For further information, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are documented in *Siebel Maintenance Release Guide* on My Oracle Support.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using and *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

- 2 Install the Siebel Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).

The upgraded Siebel Servers will not work correctly with the RDBMS server until after you have upgraded the Siebel Database to the new release.

- 3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade.
- 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*.

NOTE: You do not install a new Siebel database as part of upgrading the Siebel Enterprise.

Upgrade the RDBMS

- 6 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade. For information on supported RDBMS systems, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Identify and Resolve Duplicate Row IDs

- 7 **6.2.1 upgrades only:** [“Identifying and Resolving Duplicate Row IDs” on page 145.](#)

After you install the Siebel Database Server software, but before you upgrade to this release, you must identify and resolve any duplicate row IDs in your Siebel Database.

Preupgrade Tasks for the Siebel Database

- 8 Review Siebel Technical Notes and Siebel Alerts on upgrading to the current release. Make revisions to the upgrade process as required. See [“Important Upgrade Planning Resources” on page 83.](#)

- 9 Review 477519.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 521. It contains important changes to database records and repository objects that must be made during the upgrade. See [“Important Upgrade Planning Resources” on page 83](#).
- 10 Review guidelines for configuring the RDBMS. See *Implementing Siebel Business Applications on DB2 UDB for z/OS*.
- 11 Verify all developers have checked in their projects.
- 12 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 13 Stop the Siebel Server and the Siebel Gateway Name Server.
- 14 Verify there are no open database connections.
- 15 Perform the tasks in [Chapter 7, “Basic Database Preparations.”](#)

Preupgrade Tasks for a Development Environment Upgrade

- 16 Perform the tasks in [Chapter 9, “Preparing a Development Environment for Upgrade.”](#)

Preupgrade Tasks for Application Data

- 17 Perform the relevant tasks in [Chapter 8, “Preparing Application Data for Upgrade.”](#)

Some of these tasks are optional, depending on the currently installed Siebel products and your upgrade path. Review and perform these tasks as necessary.

Upgrade Siebel Database Schema (upgrep)

- 18 Install Siebel Tools on development workstations. Keep at least one copy of the previous version of Siebel Tools. You will need it to perform repository operations before the repository merge.
- 19 [“Migrating Repository Objects to the Standard UI” on page 271](#).
- 20 Back up the development database. (If you backed up the database as part of an RDBMS upgrade, ignore this step.)
- 21 [“Generating Development Environment Mainframe-Centric Upgrade Files” on page 202](#).
- 22 Run the Database Server Configuration utility:
 - [“Preparing to Run the Database Server Configuration Utility” on page 164](#).
 - [“Running the Database Server Configuration Utility Under Windows” on page 168](#).
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169](#).

When running the Database Server Configuration utility, choose the following settings:

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

The master UCF file is updated with the development environment configuration.

- 23 Review the following topic and perform the procedure if applicable: [“Migrating Address Data from Custom Extension Columns”](#) on page 173.
- 24 Enter Yes on the Database Server Configuration utility screen to launch the Siebel Upgrade Wizard.

SQL commands are executed on the development environment database until the Upgrade Wizard stops at the First Pause.
- 25 [“Preparing the z/OS Host Environment”](#) on page 203.
- 26 [“Resuming the Siebel Upgrade Wizard After the First Pause”](#) on page 213.
- 27 [“Transferring Control Cards and Schema DDL Files to the z/OS Host”](#) on page 215.
- 28 [“Preparing the Schema and JCL Files on the z/OS Host”](#) on page 216.
- 29 [“Installing Stored Procedures on the z/OS Host”](#) on page 219.
- 30 [“Preparing Siebel-Scheduled Jobs on the z/OS Host”](#) on page 221.
- 31 [“Optimizing the Target Job Stream, Part 1”](#) on page 224.
- 32 [“Optimizing Old-Schema Index Rebuild Jobstreams”](#) on page 225.
- 33 [“Loading Log Tables on the Source Database”](#) on page 226.
- 34 [“Loading Log Tables on the Target Database”](#) on page 227.
- 35 [“Executing the Upgrade on the z/OS Host \(Phase 1 of 2\)”](#) on page 228.
- 36 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause”](#) on page 238.
- 37 [“Transferring the Index Schema to the z/OS Host”](#) on page 239.
- 38 [“Preparing Siebel-Scheduled JCL”](#) on page 240.
- 39 [“Optimizing the Target Job Stream, Part 2”](#) on page 242.
- 40 [“Executing the Upgrade on the z/OS Host \(Phase 2 of 2\)”](#) on page 243.
- 41 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause”](#) on page 251.
- 42 [“Reviewing Upgrade Log Files for Errors”](#) on page 256.
- 43 If the upgrade contains unacceptable errors, do the following:
 - a Restore the backup of the database.
 - b Correct the errors.
 - c Rerun the Database Server Configuration utility.
- 44 [“Manually Archiving Upgrade Log Files”](#) on page 259.
- 45 [“Installing New License Keys After Upgrade”](#) on page 180.
- 46 **Multilingual deployments.** If you have multilingual deployments, perform the steps in 477094.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 447. This document describes how to import language-specific repository strings and seed data into the upgrade repositories.

- 47 Back up the upgraded database repository after a successful upgrade of the Siebel Database Schema.

Prepare for Repository Merge

- 48 Set the Upgrade Ancestor property for copied objects. See [“Configuring Objects to Inherit Upgrade Behavior” on page 270](#).
- 49 [“Migrating Repository Objects to the Standard UI” on page 271](#).
- 50 [“About Backing Up the New Customer Repository or Database Schema” on page 275](#).
- 51 **6.2.1 upgrades only:** [“Running the Repository Preparation Wizard on Release 6.2.1 Repository” on page 277](#).
- 52 Execute the REORG utility on tables that receive a large number of inserts during the repository import process. See [“About Reorganizing Tables Before the Repository Merge” on page 275](#) for further information.
- 53 Run database statistics on the Siebel Database. For further information on running database statistics, see [“Generating RUNSTATS” on page 248](#). Running statistics on the Siebel Database improves merge performance.

If upgrading from Release 7.7, run statistics specifically on the S_SYM_STR and SYM_STR_INT tables. If you are upgrading from a pre-7.7 release, the S_SYM_STR and SYM_STR_INT tables are not populated until the merge is completed so you do not need to run statistics on them at this point.

Perform Repository Merge

CAUTION: The Repository merge process cannot be stopped and restarted so ensure you have backed up the database schema or the New Customer Repository before starting the merge.

- 54 [“Performing a Repository Merge” on page 279](#).
- 55 [“Determining If a Repository Merge Was Successful” on page 288](#).
- 56 If the repository merge contains unacceptable errors, do the following:
 - a Restore the backup of the database or New Customer Repository.
 - b Correct the errors.
 - c Rerun the Database Server Configuration utility.
- 57 [“Migrating Custom Workflows” on page 289](#).
- 58 Back up the database.

Migrate 6.2.1 Customizations

6.2.1 upgrades only: The steps in [“Migrate 6.2.1 Customizations”](#) apply only to upgrades from Release 6.2.1.

- 59 [“Identifying What Will Be Migrated” on page 309](#).

- 60 “Exposing Hidden Properties of Applets and Views” on page 310.
- 61 “Running the Siebel Web Client Migration Wizard” on page 311.
- 62 “Reviewing Applet and View Migration” on page 314.
- 63 “Running the Web Layout Wizard” on page 315.
- 64 “Migrating Scripts Attached to Controls” on page 316.
- 65 “Migrating Scripts Attached to Applets” on page 318.
- 66 “Migrating Business Component, Business Service, and Application Scripts” on page 321.
- 67 “Migrating Outbound COM Interfaces” on page 322.
- 68 “Migrating Inbound COM Interfaces” on page 323.
- 69 “Regenerating the Postmerge Utilities Log” on page 326.
- 70 “Reviewing Customized Business Components” on page 326.

Perform Postmerge Tasks

- 71 “Running the Postmerge Utilities” on page 329.
- 72 (Optional.) Compile an SRF file to help review the UI. See “Creating a New SRF File” on page 297.
- 73 “Reviewing Attribute Conflicts in the Repository Merge” on page 291.
- 74 Perform the tasks in Chapter 17, “Postmerge Development Tasks.”
- 75 “Generating EIM Temporary Columns After a Repository Merge” on page 294.

Upgrade Custom Database Schema (upgphys)

- 76 Run the Database Server Configuration utility:
 - “Preparing to Run the Database Server Configuration Utility” on page 164.
 - “Running the Database Server Configuration Utility Under Windows” on page 168.
 - “Running the Database Server Configuration Utility Under UNIX” on page 169.

When running the Database Server Configuration utility, choose the following settings:

- **Upgrade Options:** Upgrade Siebel Database Schema (upgphys)
- **Environment Type:** Development

Enter Yes to launch the Siebel Upgrade Wizard. SQL commands are executed on the development environment database until the Upgrade Wizard stops at the First Pause.

- 77 “Synchronizing the Schema” on page 198.
- 78 “Dedup Files” on page 199.
- 79 “Migrating Custom Business Component Configurations” on page 199.
- 80 “Reviewing Upgrade Log Files for Errors” on page 256.

81 If the upgrade contains unacceptable errors, do the following:

- a** Restore the backup of the database.
- b** Correct the errors.
- c** Rerun the Database Server Configuration utility.

82 [“Manually Archiving Upgrade Log Files” on page 259.](#)

83 Back up the upgraded database.

84 [“Deleting Redundant Upgrade Files” on page 181.](#)

85 [“Creating a New SRF File” on page 297.](#)

Postupgrade Tasks for Environment Setup

86 [“Updating File System Attachments” on page 355.](#)

87 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide* and [“Regenerating the Database Template File” on page 348.](#)

88 Extract developers' databases. See *Siebel Remote and Replication Manager Administration Guide* and [“Extracting Developers or Siebel Tools Clients” on page 349.](#)

89 Initialize the local database on development machines.

90 [“Checking for Inactivated EIM Table Columns” on page 353.](#)

91 Reset upgrade-specific parameters back to their defaults. See [“Resetting Database Server Configuration Parameters” on page 349.](#)

92 Run database statistics. For more information, see [“Generating RUNSTATS” on page 248.](#)

NOTE: The development environment is now upgraded. The remaining sections deal with configuration and validation tasks.

Postupgrade Tasks for Configuration

93 [“Reviewing Deleted Objects in the Repository Merge” on page 292.](#)

94 [“Reviewing Obsolete Objects in the Repository Merge” on page 293.](#)

95 Resolve any business component and join conflicts.

96 [“Reapplying Custom Columns to the Siebel Database” on page 345.](#)

97 [“Deleting Duplicate EIM Mappings” on page 347.](#)

98 [“Deleting Unneeded Repository Files” on page 297.](#)

99 If you exported data from interface tables before the upgrade, review the database and import the data as desired.

100 [“Upgrading to RC2 or AES Encryption” on page 96.](#)

Configure for Globalization

- 101 Install language packs for new languages. See the *Siebel Installation Guide* for the operating system you are using.
- 102 “Upgrading to the Symbolic String Model” on page 349.
- 103 “Setting Up Your Environment to Support Global Time Zone” on page 362.

Deploy Workflows

Activate and deploy workflows. To perform these tasks, see *Siebel Business Process Designer Administration Guide*.

- 104 “Upgrading Seeded Workflows” on page 388.
- 105 “Upgrading Inbound Workflows” on page 388.

Verify Application Integration

- 106 Verify that EAI and EIM integrations are set up correctly. For information on using EAI and EIM, see *Overview: Siebel Enterprise Application Integration* and *Siebel Enterprise Integration Manager Administration Guide*.
- 107 “Updating Enterprise Application Integration (EAI) After Upgrade” on page 363.

Postupgrade Tasks for Application Administration

- 108 Review the results of the Person and Organization merge. Make configuration changes as required.
- 109 “Generating Reporting Relationships After Upgrade” on page 360.

Perform System Tests

- 110 Perform the tasks in [Chapter 21, “Postupgrade Tasks for Applications”](#) as needed to upgrade test data in the development environment.
- 111 Use available test data to perform unit testing. Validate application function in the following areas:
 - User interface
 - Data interfaces
 - Integrity of migrated data
 - Workflow function
- 112 If you revise repository objects or schema definitions, regenerate the schema.dll and custrep.dat files. See [“Regenerating the Repository Definition Files” on page 295](#).

Process of Upgrading a Production Test Environment

Upgrades: All upgrades.

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

This process is part of a roadmap. See [“Roadmap for Performing the Upgrade”](#) on page 53.

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

NOTE: The production test environment should replicate the production environment exactly.

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

Upgrade Third-Party Software

1 Upgrade third-party software as required due to dependencies on 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.

For further information, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are documented in *Siebel Maintenance Release Guide* on My Oracle Support.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using and *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

2 Install the Siebel Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).

The upgraded Siebel Servers will not work correctly with the RDBMS server until after you have upgraded the Siebel Database to the new release.

3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade.

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*.

NOTE: You do not install a new Siebel database as part of upgrading the Siebel Enterprise.

Upgrade the RDBMS

- 6 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade. For information on supported RDBMS systems, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Identify and Resolve Duplicate Row IDs

- 7 **6.2.1 upgrades only:** [“Identifying and Resolving Duplicate Row IDs” on page 145.](#)

After you install the Siebel Database Server software, but before you upgrade to this release, you must identify and resolve any duplicate row IDs in your Siebel Database.

Preupgrade Tasks for the Siebel Database

- 8 Review Siebel Technical Notes and Siebel Alerts on upgrading to the current release. Make revisions to the upgrade process as required. See [“Important Upgrade Planning Resources” on page 83.](#)
- 9 Review 477519.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 521. It contains important changes to database records and repository objects that must be made during the upgrade. See [“Important Upgrade Planning Resources” on page 83.](#)
- 10 Review guidelines for configuring the RDBMS. See *Implementing Siebel Business Applications on DB2 UDB for z/OS*.
- 11 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 12 Stop the Siebel Server and the Siebel Gateway Name Server.
- 13 Verify there are no open database connections.
- 14 Perform the tasks in [Chapter 7, “Basic Database Preparations.”](#)

Preupgrade Tasks for a Production Environment Upgrade

- 15 Perform the tasks in [Chapter 10, “Preparing a Production Environment for Upgrade.”](#)

You must perform the task [“Transferring the Customized Repository and Schema Definition Files” on page 161.](#)

Preupgrade Tasks for Application Data

- 16 Perform the relevant tasks in [Chapter 8, “Preparing Application Data for Upgrade.”](#)

Some of these tasks are optional, depending on the currently installed Siebel products and your upgrade path. Review and perform these tasks as necessary.

Prepare the Siebel Database for Upgrade

- 17 If you revised repository objects or schema definitions after performing your development environment upgrade, regenerate the schema.ddl and custrep.dat files and import them to the production test environment. See [“Regenerating the Repository Definition Files” on page 295](#) and [“Transferring the Customized Repository and Schema Definition Files” on page 161](#).
- 18 Verify the production test database is either a copy of the current production database or has the same topology and a similar amount of data. This is important for effective performance testing of the upgrade scripts.
- 19 Verify you have a current backup of the production test environment database. To do upgrade tuning, you will restore this database and perform test-upgrades on it.
- 20 On the Siebel Server you used to upgrade the development environment database, create an ODBC to connect to the production test environment database.
- 21 Disconnect the Siebel Server from the development environment database and connect it to the production test environment database.
- 22 Run the Database Server Configuration utility:
 - [“Preparing to Run the Database Server Configuration Utility” on page 164](#).
 - [“Running the Database Server Configuration Utility Under Windows” on page 168](#).
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169](#).

Choose the following settings:

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

Upgrade the Siebel Database Schema (upgrep + upgphys)

- 23 Run the Database Server Configuration utility:
 - [“Preparing to Run the Database Server Configuration Utility” on page 164](#).
 - [“Running the Database Server Configuration Utility Under Windows” on page 168](#).
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169](#).

When running the Database Server Configuration utility, choose the following settings:

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

Enter configuration information for the production test environment. The master UCF file is updated with the production test environment configuration.

- 24 Review the following topic and perform the procedure if applicable: [“Migrating Address Data from Custom Extension Columns” on page 173](#).
- 25 Enter Yes on the Database Server Configuration utility screen to launch the Siebel Upgrade Wizard. SQL commands are executed on the production test environment database until the Upgrade Wizard stops at the First Pause.

- 26 "Preparing the z/OS Host Environment" on page 203.
- 27 "Resuming the Siebel Upgrade Wizard After the First Pause" on page 213.
- 28 "Transferring Control Cards and Schema DDL Files to the z/OS Host" on page 215.
- 29 "Preparing the Schema and JCL Files on the z/OS Host" on page 216.
- 30 "Installing Stored Procedures on the z/OS Host" on page 219.
- 31 "Preparing Siebel-Scheduled Jobs on the z/OS Host" on page 221.
- 32 "Optimizing the Target Job Stream, Part 1" on page 224.
- 33 "Optimizing Old-Schema Index Rebuild Jobstreams" on page 225.
- 34 "Loading Log Tables on the Source Database" on page 226.
- 35 "Loading Log Tables on the Target Database" on page 227.
- 36 "Executing the Upgrade on the z/OS Host (Phase 1 of 2)" on page 228.
- 37 "Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause" on page 238.
- 38 "Transferring the Index Schema to the z/OS Host" on page 239.
- 39 "Preparing Siebel-Scheduled JCL" on page 240.
- 40 "Optimizing the Target Job Stream, Part 2" on page 242.
- 41 "Executing the Upgrade on the z/OS Host (Phase 2 of 2)" on page 243.
- 42 "Performing Intersection Table Maintenance" on page 249.
- 43 "Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause" on page 251.
- 44 "Reviewing Upgrade Log Files for Errors" on page 256.
- 45 If the upgrade contains unacceptable errors, do the following:
 - a Restore the backup of the database.
 - b Correct the errors.
 - c Rerun the Database Server Configuration utility.
- 46 "Manually Archiving Upgrade Log Files" on page 259.
- 47 "Installing New License Keys After Upgrade" on page 180.
- 48 Back up the upgraded database.
- 49 "Deleting Redundant Upgrade Files" on page 181.

Postupgrade Tasks for Environment Setup

- 50 "Updating File System Attachments" on page 355.
- 51 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide* and "Upgrading Siebel Mobile and Dedicated Web Clients" on page 351.
- 52 Generate database extracts for all Regional Servers in the deployment. See "Upgrading Regional Servers" on page 351.

53 Reset upgrade-specific parameters back to their defaults. See [“Resetting Database Server Configuration Parameters”](#) on page 349.

54 [“Checking for Inactivated EIM Table Columns”](#) on page 353.

55 Run database statistics. For more information, see [“Generating RUNSTATS”](#) on page 248.

NOTE: The production test environment is now upgraded. The remaining sections deal with configuration and validation tasks.

Postupgrade Tasks for Configuration

56 If you exported data from interface tables before the upgrade, review the database and import the data as desired.

57 [“Upgrading to RC2 or AES Encryption”](#) on page 96.

Configure for Globalization

58 [“Upgrading to the Symbolic String Model”](#) on page 349.

59 [“Setting Up Your Environment to Support Global Time Zone”](#) on page 362.

Deploy Workflows

Activate and deploy workflows. To perform these tasks, see *Siebel Business Process Designer Administration Guide*.

60 [“Upgrading Seeded Workflows”](#) on page 388.

61 [“Upgrading Inbound Workflows”](#) on page 388.

Verify Application Integration

62 Verify that EAI and EIM integrations are set up correctly. For information on using EAI and EIM, see *Overview: Siebel Enterprise Application Integration* and *Siebel Enterprise Integration Manager Administration Guide*.

63 [“Updating Enterprise Application Integration \(EAI\) After Upgrade”](#) on page 363.

Postupgrade Tasks for Application Administration

64 Review the results of the Person and Organization merge. Make configuration changes as required.

65 [“Generating Reporting Relationships After Upgrade”](#) on page 360.

66 [“Validating Dock Objects and Rule Definitions”](#) on page 353.

67 Perform the tasks in [Chapter 21, “Postupgrade Tasks for Applications”](#) to prepare for system testing.

Perform System Tests

68 Thoroughly test all applications.

Process of Tuning Upgrade Performance

Upgrades: All upgrades.

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

This process is optional.

This process is part of a roadmap. See [“Roadmap for Performing the Upgrade”](#) on page 53.

CAUTION: You *must* contact Siebel Expert Services before tuning upgrade scripts. If you do not, you may invalidate your support agreement.

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. For help with this process and to implement more advanced tuning, contact Siebel Expert Services.

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

- 478308.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 616. This document describes strategies for minimizing production environment downtime during an upgrade. The steps in this process are intended primarily for use with the Baseline best practices described in Technical Note 616.
- [Chapter 22, “Tuning the Production Upgrade Scripts.”](#) This chapter provides information on how you can improve the performance of the production environment upgrade by tuning the production upgrade scripts in a production test environment.

Set Up the Target Database

- 1 Back up and remove the upgraded production test 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 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 is configured for optimum upgrade performance. See [Chapter 7, “Basic Database Preparations.”](#)
- 5 Run statistics on the target database. This optimizes query performance.
- 6 Perform the tasks in [Chapter 8, “Preparing Application Data for Upgrade.”](#)

Upgrade the Target Database Schema (upgrep + upgphys)

7 In the production test environment, run the Database Server Configuration Utility:

- [“Preparing to Run the Database Server Configuration Utility” on page 164.](#)
- [“Running the Database Server Configuration Utility Under Windows” on page 168.](#)
- [“Running the Database Server Configuration Utility Under UNIX” on page 169.](#)

Choose the following settings:

- **Upgrade Options:** Upgrade Siebel Database Schema (upgrep + upgphys)
- **Environment Type:** Production test.

Enter Yes to launch the Siebel Upgrade Wizard. SQL commands are executed on the production test environment database until the Upgrade Wizard stops at the First Pause.

8 Review the following topic and perform the procedure if applicable: [“Migrating Address Data from Custom Extension Columns” on page 173.](#)

9 [“Preparing the z/OS Host Environment” on page 203.](#)

10 [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 213.](#)

11 [“Transferring Control Cards and Schema DDL Files to the z/OS Host” on page 215.](#)

12 [“Preparing the Schema and JCL Files on the z/OS Host” on page 216.](#)

13 [“Installing Stored Procedures on the z/OS Host” on page 219.](#)

14 [“Preparing Siebel-Scheduled Jobs on the z/OS Host” on page 221.](#)

15 [“Optimizing the Target Job Stream, Part 1” on page 224.](#)

16 [“Optimizing Old-Schema Index Rebuild Jobstreams” on page 225.](#)

17 [“Loading Log Tables on the Source Database” on page 226.](#)

18 [“Loading Log Tables on the Target Database” on page 227.](#)

19 [“Executing the Upgrade on the z/OS Host \(Phase 1 of 2\)” on page 228.](#)

20 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause” on page 238.](#)

21 [“Transferring the Index Schema to the z/OS Host” on page 239.](#)

22 [“Preparing Siebel-Scheduled JCL” on page 240.](#)

23 [“Optimizing the Target Job Stream, Part 2” on page 242.](#)

24 [“Executing the Upgrade on the z/OS Host \(Phase 2 of 2\)” on page 243.](#)

25 [“Performing Intersection Table Maintenance” on page 249.](#)

26 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause” on page 251.](#)

Note the time required to upgrade the database.

27 Review the upgrade logs for errors. See [“Reviewing Upgrade Log Files for Errors” on page 256.](#)

28 If the upgrade contains errors that prevented completion or adversely affected performance, correct the errors and rerun the upgrade.

Tune the Upgrade Files

- 29** Evaluate upgrade performance, particularly the time required to complete the upgrade.
- 30** Do one of the following:
- If the time required to complete the upgrade is acceptable, no further tuning is needed.
 - If the time required to complete the upgrade is too long, perform the remaining steps in this section to continue tuning upgrade performance.
 - If the time required to complete the upgrade is too long and you cannot tune further, contact Siebel Expert Services to apply advanced tuning.
- 31** Carefully review target database and RDBMS server configuration. Adjust as needed to further improve upgrade performance.
- 32** Tune the upgrade files:
- [“Optimizing Unload and Load Job Performance” on page 392.](#)
 - [“Adding the Statistics Clause to Load Cards” on page 392.](#)
 - [“About Deactivating Jobs That Do Not Process Data” on page 393.](#)
- 33** Copy the tuned upgrade scripts to a safe location for use in the production upgrade.

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.

- 34** 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.
- 35** In the production test environment, perform another test upgrade and evaluate upgrade performance.
- 36** Repeat the tuning process and perform test-upgrades until upgrade performance is acceptable.
- 37** When you have completed tuning upgrade performance in the production test environment, delete and remove the target database.

Process of Upgrading a Production Environment

Upgrades: All upgrades.

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

This process is part of a roadmap. See [“Roadmap for Performing the Upgrade” on page 53.](#)

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.

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

Upgrade Third-Party Software

1 Upgrade third-party software as required due to dependencies on 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.

For further information, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrade the Servers

Verify you have identified all the maintenance releases, Fix Packs, and quick-fix patches required for the upgrade. These requirements are documented in *Siebel Maintenance Release Guide* on My Oracle Support.

To perform the following steps, see the *Siebel Installation Guide* for the operating system you are using and *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

2 Install the Siebel Gateway Name Server, Siebel Servers, and Siebel Web Server Extension (SWSE).

The upgraded Siebel Servers will not work correctly with the RDBMS server until after you have upgraded the Siebel Database to the new release.

3 Install the Siebel Database Server files on the Siebel Server you will use to perform the upgrade.

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*.

NOTE: You do not install a new Siebel database as part of upgrading the Siebel Enterprise.

Upgrade the RDBMS

6 If required, upgrade the RDBMS version. Refer to the vendor's documentation to perform the upgrade. For information on supported RDBMS systems, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Identify and Resolve Duplicate Row IDs

7 **6.2.1 upgrades only:** [“Identifying and Resolving Duplicate Row IDs” on page 145.](#)

After you install the Siebel Database Server software, but before you upgrade to this release, you must identify and resolve any duplicate row IDs in your Siebel Database.

Preupgrade Tasks for the Siebel Database

- 8 Review Siebel Technical Notes and Siebel Alerts on upgrading to the current release. Make revisions to the upgrade process as required. See [“Important Upgrade Planning Resources” on page 83](#).
- 9 Review 477519.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 521. It contains important changes to database records and repository objects that must be made during the upgrade. See [“Important Upgrade Planning Resources” on page 83](#).
- 10 Review guidelines for configuring the RDBMS. See *Implementing Siebel Business Applications on DB2 UDB for z/OS*.
- 11 Verify that the Workflow Monitor and Workflow action agents have processed all pending requests.
- 12 Stop the Siebel Server and the Siebel Gateway Name Server.
- 13 Verify there are no open database connections.
- 14 Perform the tasks in [Chapter 7, “Basic Database Preparations.”](#)

Preupgrade Tasks for a Production Environment Upgrade

- 15 Perform the relevant tasks in [Chapter 10, “Preparing a Production Environment for Upgrade.”](#)
You must perform the task [“Transferring the Customized Repository and Schema Definition Files” on page 161](#).

Preupgrade Tasks for Application Data

- 16 Perform the relevant tasks in [Chapter 8, “Preparing Application Data for Upgrade.”](#)

Some of these tasks are optional, depending on the currently installed Siebel products and your upgrade path. Review and perform these tasks as necessary.

Upgrade the Siebel Database Schema (upgrep + upgphys)

- 17 Verify you have a current backup of the production environment database.
- 18 On the Siebel Server you used to upgrade the production test environment, create an ODBC to connect to the production environment database.
- 19 Run the Database Server Configuration utility:
 - [“Preparing to Run the Database Server Configuration Utility” on page 164](#).
 - [“Running the Database Server Configuration Utility Under Windows” on page 168](#).
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169](#).

When running the Database Server Configuration utility, choose the following settings:

- **Upgrade Options:** Upgrade Siebel Database Schema (upgrep + upgphys)

■ Environment Type: Production

You do not have to run the Database Server Configuration Utility in Prepare for Production mode.

NOTE: If you have completed a production test upgrade and have tuned the upgrade scripts used to perform the mainframe upgrade tasks, ensure that you copy these files to a safe location so that you can use them during the production upgrade. This will ensure you do not overwrite them when you next run the Database Server Configuration utility.

- 20 Enter configuration information for the production environment. The master UCF file is updated with the production environment configuration.
- 21 Review the following topic and perform the procedure if applicable: [“Migrating Address Data from Custom Extension Columns” on page 173.](#)
- 22 Enter Yes on the Database Server Configuration utility screen to launch the Siebel Upgrade Wizard. SQL commands are executed on the production database until the Upgrade Wizard stops at the First Pause.

NOTE: If you have completed a production test upgrade and have generated and tuned the SQL and JCL upgrade scripts that are run on the mainframe, you may not have to perform all of the tasks listed in the rest of this section. Instead of transferring the files generated by the Upgrade Wizard to the mainframe, use the files that you generated and tuned during the Production Test environment upgrade to perform the mainframe upgrade tasks. Then resume the Upgrade Wizard on the midtier so that it can perform the midtier upgrade tasks. Siebel Expert Services can advise on which of the following steps you must run; this may depend on the upgrade script tuning that you have performed.

- 23 [“Preparing the z/OS Host Environment” on page 203.](#)
- 24 [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 213.](#)
- 25 [“Transferring Control Cards and Schema DDL Files to the z/OS Host” on page 215.](#)
- 26 [“Preparing the Schema and JCL Files on the z/OS Host” on page 216.](#)
- 27 [“Installing Stored Procedures on the z/OS Host” on page 219.](#)
- 28 [“Preparing Siebel-Scheduled Jobs on the z/OS Host” on page 221.](#)
- 29 [“Optimizing the Target Job Stream, Part 1” on page 224.](#)
- 30 [“Optimizing Old-Schema Index Rebuild Jobstreams” on page 225.](#)
- 31 [“Loading Log Tables on the Source Database” on page 226.](#)
- 32 [“Loading Log Tables on the Target Database” on page 227.](#)
- 33 [“Executing the Upgrade on the z/OS Host \(Phase 1 of 2\)” on page 228.](#)
- 34 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause” on page 238.](#)
- 35 [“Transferring the Index Schema to the z/OS Host” on page 239.](#)
- 36 [“Preparing Siebel-Scheduled JCL” on page 240.](#)
- 37 [“Optimizing the Target Job Stream, Part 2” on page 242.](#)
- 38 [“Executing the Upgrade on the z/OS Host \(Phase 2 of 2\)” on page 243.](#)

- 39 [“Performing Intersection Table Maintenance” on page 249.](#)
- 40 [“Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause” on page 251.](#)
- 41 [“Reviewing Upgrade Log Files for Errors” on page 256.](#)
- 42 If the upgrade contains unacceptable errors, do the following:
 - a Restore the backup of the database.
 - b Correct the errors.
 - c Rerun the Database Server Configuration utility.
- 43 [“Manually Archiving Upgrade Log Files” on page 259.](#)
- 44 [“Installing New License Keys After Upgrade” on page 180.](#)
- 45 Back up the upgraded database.
- 46 [“Deleting Redundant Upgrade Files” on page 181.](#)

Postupgrade Tasks for Environment Setup

- 47 [“Updating File System Attachments” on page 355.](#)
- 48 Generate a Siebel Remote database template file. See *Siebel Remote and Replication Manager Administration Guide* and [“Upgrading Siebel Mobile and Dedicated Web Clients” on page 351.](#)
- 49 Generate database extracts for all Regional Servers in the deployment. See [“Upgrading Regional Servers” on page 351.](#)
- 50 Reset upgrade-specific parameters back to their defaults. See [“Resetting Database Server Configuration Parameters” on page 349.](#)
- 51 [“Checking for Inactivated EIM Table Columns” on page 353.](#)
- 52 Run database statistics. For more information on running statistics, see [“Generating RUNSTATS” on page 248.](#)

NOTE: The production environment is now upgraded. The remaining sections deal with configuration and validation tasks.

Postupgrade Tasks for Configuration

- 53 If you exported data from interface tables before the upgrade, review the database and import the data as desired.
- 54 [“Upgrading to RC2 or AES Encryption” on page 96.](#)

Configure for Globalization

- 55 [“Upgrading to the Symbolic String Model” on page 349.](#)
- 56 [“Setting Up Your Environment to Support Global Time Zone” on page 362.](#)

Deploy Workflows

Activate and deploy workflows. To perform these tasks, see *Siebel Business Process Designer Administration Guide*.

- 57 “Upgrading Seeded Workflows” on page 388.
- 58 “Upgrading Inbound Workflows” on page 388.

Verify Application Integration

- 59 Verify that EAI and EIM integrations are set up correctly. For information on using EAI and EIM, see *Overview: Siebel Enterprise Application Integration* and *Siebel Enterprise Integration Manager Administration Guide*.
- 60 “Updating Enterprise Application Integration (EAI) After Upgrade” on page 363.

Postupgrade Tasks for Application Administration

- 61 Review the results of the Person and Organization merge. Make configuration changes as required.
- 62 “Generating Reporting Relationships After Upgrade” on page 360.
- 63 “Validating Dock Objects and Rule Definitions” on page 353.
- 64 Perform the tasks in Chapter 21, “Postupgrade Tasks for Applications” to prepare for system testing.

Perform System Tests

- 65 Thoroughly test all applications.

Deploy to Users

- 66 If you have customized the configuration of Siebel Enterprise Server components, such as Siebel Servers, you must manually enter the customizations in the upgraded environments. See *Going Live with Siebel Business Applications*.
- 67 Use the Siebel Application Deployment Manager to migrate administrative data such as LOVs and responsibilities from development to production. See *Going Live with Siebel Business Applications*.
- 68 Use Siebel Packager to create language packs for remote installation. See *Going Live with Siebel Business Applications*.
- 69 Use Siebel Anywhere to create installation kits for deployment. See *Siebel Anywhere Administration Guide*.
- 70 Generate a Siebel Remote database template. See *Siebel Remote and Replication Manager Administration Guide*.
- 71 Set up database extraction for Siebel Mobile Web Clients. See *Siebel Remote and Replication Manager Administration Guide*.

5

Database and UI Upgrade Planning

This chapter describes the important database-related issues you need to consider when planning an upgrade to Siebel 7.8. It includes the following topics:

- “Important Upgrade Planning Resources”
- “Best Practices for Doing Your Upgrade” on page 85
- “Planning Your Upgrade” on page 87
- “About Estimating Database Size” on page 94
- “Upgrading Your DB2 Software” on page 95
- “About Multilingual Deployments” on page 95
- “Upgrading to RC2 or AES Encryption” on page 96
- “New Upgrade Features” on page 96
- “About User Interface Changes” on page 97
- “About Upgrading Access Control” on page 98
- “About the Party Model” on page 99
- “About the New Column Added to S_SRC_PAYMENT” on page 101
- “About Database Clustered Indexes” on page 102
- “About Migrating HTML Attachments to Base Tables” on page 102

Important Upgrade Planning Resources

Upgrades: All upgrades.

Environments: All environments.

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 Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>) and Oracle Software Delivery Cloud. It might also be installed locally on your intranet or on a network location.

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*:

- *Implementing Siebel Business Applications on DB2 UDB for z/OS*
- *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* for information about how to migrate customizations from the development environment to the production environment
- *Siebel System Administration Guide* for details on how to administer, maintain, and expand your Siebel Servers
- *Security Guide for Siebel Business Applications*
- *Performance Tuning Guide*
- *Configuring Siebel Business Applications* for information about configuring Siebel Business Applications in Siebel Tools
- **6.2.1 upgrades only:** *Planning an Upgrade to Siebel 7* for information about upgrading to the Siebel 7 data model for the first time
- **6.2.1 upgrades only:** *Siebel Connector for SAP/R3* to obtain Siebel 6.2.1 upgrade instructions specific to this connector

Oracle Technology Network

This site contains the following:

- ***Siebel Bookshelf***. A searchable collection of Oracle's Siebel product documentation.
- ***Siebel System Requirements and Supported Platforms on Oracle Technology Network***. This document is the definitive list of system requirements and supported third-party products.

My Oracle Support

This site provides search access to Bulletins (previously published as Siebel Technical Notes) Alerts, Fix Packs, Release Notes, Maintenance Release Guides, and more:

- ***Release Notes***. *Siebel Release Notes* on My Oracle Support contain late-breaking information that this guide does not yet include.
- ***Maintenance Release Guides***. Maintenance Release Guides contain important information about updates to applications in maintenance releases. See the applicable *Siebel Maintenance Release Guide* on My Oracle Support.

- **Bulletins and Alerts.** Bulletins and Alerts provide important information on specific upgrade issues. Bulletins related to upgrade issues are located on My Oracle Support.
477833.1 (Article ID) on My Oracle Support (previously published as Siebel Alert 1002) and 478177.1 (Article ID) on My Oracle Support (previously published as Siebel Alert 1179) are master Alerts particularly important to all upgrades. References to these and other Alerts and Bulletins are integrated throughout the guide.
- **Troubleshooting.** Troubleshooting documents contain information about how to troubleshoot common upgrade issues such as error messages encountered during the upgrade process, and other unwanted behavior in Oracle's Siebel applications. Troubleshooting documents can be found for a variety of upgrade issues, including error messages found in upgrade logs. For more information, browse troubleshooting content on My Oracle Support.

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 Siebel resources.

Siebel Expert Services

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. For further information, see ["About Using Siebel Expert Services" on page 26](#).

Best Practices for Doing Your Upgrade

Upgrades: All upgrades.

Environments: All environments.

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

Use the following steps to help plan your upgrade.

- 1 **Determine your upgrade path.** First refer to *Siebel System Requirements and Supported Platforms* on Oracle Technology Network 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 Guide* on My Oracle Support for the release to which you are upgrading. The *Siebel Maintenance Release Guide* on My Oracle Support lists the upgrade path from the major release to its most recent maintenance release. Maintenance releases roll up fixpacks and also include new features. Because fixes to the upgrade process may exist in fixpacks 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 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 Siebel Systems 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 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. See [Chapter 22, “Tuning the Production Upgrade Scripts”](#) for further information.
NOTE: If you intend tuning your upgrade scripts, you *must* contact Siebel Expert Services before performing this task.
- 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 Siebel 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 Siebel software
- Strict adherence to industry best practices and best practices identified in this guide

Siebel Systems recommends that you engage Siebel Expert Services for help in planning your upgrade on DB2 for z/OS. Siebel Expert Services can help you to:

- Take advantage of the new features provided by Release 7.8
- Customize the upgrade scripts and the upgrade process as appropriate for your installation
- Carry out performance tuning on the upgrade scripts to minimize production downtime

The upgrade planning process should produce a roadmap 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 *Siebel System Requirements and Supported Platforms* on Oracle Technology Network, and review Release Notes, Maintenance Release Guides, and Siebel Alerts related to upgrades to verify your upgrade path.
- Gather all relevant documentation that describes the current implementation, for example requirements documents, design documents, and architecture context diagrams.
- Do not start a new development effort until after the new version has been rolled out. This is especially important when you are upgrading from Release 6.2.1 to Release 7.8.x. There are significant user interface differences between the two versions, and you must change existing functionality before moving on to customized functionality.
- 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.

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 preupgrade data migration is required. Determine what postupgrade schema changes are required. For a list of tables that are affected by the upgrade, see [Appendix A, "Tables Modified or Seeded During Upgrade."](#)
- Consider database layout in your planning. Plan to tune the database and Siebel Database Server for the upgrade, because settings and parameters for upgrade differ from those required for online transaction processing (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. 477094.1 (Article ID) on My Oracle Support has information on upgrading multilingual deployments to Siebel 7.5.x and later. This document was previously published as Siebel Technical Note 447.
- Consider increasing the size of your tablespaces before going live. Make sure that your custom tablespaces are large enough for upgraded tables.

Planning Your Upgrade

Upgrades: All upgrades.

Environments: All environments.

There are several ways that you can proceed with the Siebel 7.8 upgrade. Consider your schedule, requirements, and available resources as you perform the following Siebel 7.8 upgrade tasks.

- [“Planning Changes to the Physical Layout of the Schema”](#)
- [“Testing Before a Production Upgrade” on page 90](#)
- [“Considering Code Page Support” on page 90](#)
- [“Executions of Jobs Using Siebel-Scheduled Mode or Vendor-Scheduled Mode” on page 91](#)
- [“Source and Target Database Planning” on page 91](#)
- [“Obtaining Required Software and Hardware” on page 92](#)
- [“Obtaining Required IBM Utilities” on page 92](#)
- [“Obtaining Required Security Privileges” on page 92](#)
- [“Planning Backup and Recovery Stages” on page 93](#)
- [“Creating a Schedule for the Upgrade” on page 94](#)

Planning Changes to the Physical Layout of the Schema

In planning your upgrade, you must understand the existing physical layout of your schema and determine whether or not you need to change this for the upgrade to Siebel 7.8. You should also consider database space requirements and whether or not you need to move tablespaces. These issues are discussed in this topic.

New Database Schema Structure Since Release 7.7

In Release 7.7, Siebel introduced the current database schema structure, referred to as the *1:1:1* model. It has the following characteristics:

- One table in each tablespace
- One tablespace in each database

Prior to Release 7.7, the database schema was built using approximately 20 databases, each of which contained multiple tablespaces. Each of these tablespaces (if non-partitioned) contained multiple tables. Release 7.8 contains thousands of databases. For example, an SIA installation has approximately 2700 databases. Each database has one tablespace and each tablespace has one table.

This model meets IBM recommendations and prevents database descriptor (DBD) locking and logging. These issues arise due to the increasing intensity of DB2 DML and DDL operations and the interaction of these operations with the DBD. The DBD is locked when information about the DB2 objects contained by the DBD is requested and accessed. In general, the more objects a DBD contains, the more probable that a DBD lock will be requested when information about a child object of the DBD is accessed.

Locks are acquired on the DBD table space (DBD01) if a DBD is not in memory (EDM pool). If the DBD is in the EDM pool, no lock is acquired on it if the SQL being run is static. However, most SQL executed by Siebel applications is dynamic; this means locks are acquired on the DBD. For more information on DBD locking, refer to the relevant IBM documentation.

The adoption of the *1:1:1* model since Release 7.7.x means that you must decide how much of this model to deploy. You have the following options:

- Create all tables in the *1:1:1* model.

New 7.8 tables are created in the *1:1:1* model and the Siebel supplied storage control file is used for the upgrade.

- Create new tables in the *1:1:1* model and maintain existing tables in their current tablespace if possible.

The following scenarios arise if you select this latter option:

- Some existing tables need to be moved to incorporate the addition of new columns.
- Some existing tables need to be moved, because they have been extended and the addition of new columns causes the table's LRECL to exceed that of the tablespace. This necessitates the use of the extract and merge methodology to create the storage control file. For more information on this methodology, see [“Extracting the Storage Control File” on page 122](#).
- Tables that are to retain the existing format are *merged* into the template control file which employs the *1:1:1* model.

For both options, enter existing tables that are to be migrated to the *1:1:1* model in the file `override.inp`. See [“About the Override File” on page 48](#) for further information.

For more information on using storage control files, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

Preparing a Storage Control File

A key task for a successful upgrade is the building of a suitable storage control file for both the development and production upgrade. You must consider space requirements. This is particularly important for the development upgrade, because three new repositories are imported into the database (one extra repository is imported during the production upgrade). Some repository tables will increase significantly in size, so you must provide sufficient space for expected database growth. See [“Preparing the Storage Layout of the Schema” on page 119](#) for more information.

About Moving Tablespaces

If you want to move tables from one tablespace to another, you need to recreate the tables in the new tablespace and then drop the existing tablespace. You cannot change the bufferpool designation in the storage control file to move tables because the page size is associated with the tablespace.

For example, if you are making changes to an existing tablespace that is using BP1 or a 4K bufferpool and these changes cause you to receive a warning from ddlimp that the table will now need to be in a 16K bufferpool, you cannot just change the bufferpool designation in the storage control file from BP1 to BP16K1. If you change the bufferpool designation, ddlimp will generate the following statement:

```
ALTER TABLESPACE <NAME> BUFFERPOOL PB16K1
```

This command will fail when executed because it is not permissible to alter the bufferpool associated with the tablespace unless the new bufferpool designation is the same size as the existing bufferpool designation.

Testing Before a Production Upgrade

Careful testing is critical for a successful upgrade. In particular, the production upgrade must be thoroughly tested to avoid data-specific issues and gain the best possible performance during your upgrade.

NOTE: Do not go live on Siebel 7.8 without exhaustive performance testing.

Considering Code Page Support

Siebel Systems supports both ASCII- and EBCDIC-based coded character set IDs (CCSIDs) on DB2 for z/OS. Development databases, however, require ASCII code pages, because databases with EBCDIC code pages do not support two critical procedures in a development environment upgrade:

- Merging prior configuration changes into a new custom configuration repository
- Compiling a new Siebel repository file (.srf file) from the new repository

Additional limitations on databases with EBCDIC code pages include the following:

- Siebel Web Client migration is not supported
- Siebel Dun & Bradstreet server components are not supported

Before you conduct an upgrade, carefully read *Siebel Release Notes* on My Oracle Support for information about known restrictions. For guidelines about choosing the code page for your subsystem, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

About Code Page Conversion

Siebel Systems supports the ASCII 5348 and EBCDIC 1140 code pages on DB2 for z/OS (see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network for further information). If your current Siebel release runs on a different code page, you must migrate to one of the supported code pages before upgrading to Siebel 7.8.

Siebel applications that are upgraded to Siebel v7.7.x or later, and that run on unsupported code pages, do not support all the characters that were supported before the upgrade, for example, the euro symbol.

Ensure you schedule sufficient time to perform and test the code page conversion before beginning the upgrade. For advice and assistance on converting DB2 code pages, contact IBM.

Executions of Jobs Using Siebel-Scheduled Mode or Vendor-Scheduled Mode

Before you perform a production upgrade, determine whether you will execute the jobs using Siebel-provided job scheduling or a third-party vendor scheduler. Choose your scheduling mode carefully, because once you begin an upgrade process under a selected mode, you cannot change your scheduling mode or reverse this decision.

In Siebel Scheduled Mode, the Siebel job scheduler uses job submission EXECs to run the upgrade jobs and automatically submits dependent jobs by their predecessors. If you plan to use a third-party job scheduler, you will find it useful to first generate the upgrade scripts using Siebel scheduling to gain an understanding of job dependencies.

Using Siebel Scheduled Mode, if a job ends abnormally or returns an invalid return code, the upgrade process is halted. You can check the job status by querying the Siebel job log. For further information, see [Chapter 14, "Reviewing the Database Upgrade Log Files."](#)

NOTE: Siebel logging is not available if you use a third-party job scheduler.

Source and Target Database Planning

For recovery purposes, in-place mainframe-centric upgrades are not supported on the z/OS host. A target database is built for the new release and data is unloaded from the existing source database and migrated to the upgraded database.

During the planning process, you must decide whether the source and target databases are to be in different DB2 subsystems or in the same subsystem. If they are located in one DB2 subsystem, the databases must have different tableowner names and database prefixes. If the target and source databases are in different DB2 subsystems, one *or* two database prefixes and tableowner names can be used.

If you want to keep both source and target databases, you must plan for additional disk space. Verify that you have enough space on the z/OS host to allow for the unloading of datasets. Typically, you need at least 200 cylinders of space.

DB2 DSNZPARM Settings For the Target Database

For Siebel Business Applications to run correctly and efficiently, ensure that the DSNZPARM parameters for your target database are set correctly. For a list of the Siebel required and recommended DSNZPARM parameter settings for DB2 for z/OS v7 and DB2 for z/OS v8, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

Obtaining Required Software and Hardware

Because each enterprise has specific needs for hardware and software resources, it is recommended that you discuss your particular situation with your Siebel technical resource. Make sure your hardware supports the needs of your Siebel 7.8 upgrade.

CAUTION: Read *Siebel System Requirements and Supported Platforms on Oracle Technology Network* to verify the supported computer and operating system platforms and supported third-party programs for this release of your Siebel Business applications.

Obtaining Required IBM Utilities

Verify that the following IBM utilities are available for your upgrade to Siebel 7.8 for DB2 UDB for z/OS:

- **DSNTIAUL.** The upgrade uses DSNTIAUL to unload data delivered by Siebel.
Prepare and bind DSNTIAUL by following the procedures in your IBM installation documentation. If you made local modifications, you might need to prepare and bind a separate version.
- **DSNTEP2.** The upgrade uses DSNTEP2 to execute SQL.
Prepare and bind DSNTEP2 by following the procedures in your IBM installation documentation. If you made local modifications, you might need to prepare and bind a separate version. Also, if you are using a separate version of DSNTEP2, you need to change the SIEBSQL* members to reflect the new plan and program names.
- **DFSORT.** The utility DFSORT is used to manipulate data for data migration during upgrade.
- **LOAD.** The IBM DB2 Load utility is used to load data during data migration.
- **REBUILD INDEX.** The IBM DB2 REBUILD INDEX utility is used to build indexes after they are created using DEFER YES.
- **IEBCOPY.** The utility IEBCOPY is used to create members in installation datasets. Sequential datasets contains control information used by IEBCOPY.
- **IEBGENER.** The utility IEBGENER is used to copy sequential datasets.

For information about IBM utilities, see your IBM documentation.

CAUTION: You can choose to use alternate third-party utility products that are preferred for your environment. Evaluate utilities by individual job. Be aware, however, that if you do use utilities other than those recommended above, you might have to modify the Siebel-supplied SQL to accord with the rules for those utilities. You must contact Siebel Expert Services for help with this task.

Obtaining Required Security Privileges

For detailed information about security for DB2 UDB for z/OS installations and upgrades, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

In Siebel 7, access privileges to database resources such as tables, views, and triggers are granted to a user group. A user group is a definition within the security package (for example, RACF) that has a common set of users attached to it. Access to the DB2 tables is granted to the user group, and user authentication is performed at the group level. All users belonging to the group are allowed access. All users that are not part of the group are denied access.

The user who executes the upgrade must be a member of a qualified group. To grant this user tableowner privileges, the tableowner must be set up as a qualified group, and the DBA who executes DDLs must be a member of this qualified group. The group ID is the qualifier (for example, RACF group ID).

The Siebel installation process allows the installer to specify the group user name for client access (the default is SSEROLE), and the resulting installation scripts generate the appropriate GRANT statements. GRANT statements for additional security groups that may be required must be created manually.

NOTE: The GRANT statements must be executed by either the tableowner, a database administrator, or a system administrator.

The following privileges are necessary for the user who performs the upgrade:

- Read the DB2 catalog
- Execute stored procedures
- Bind stored procedures

Because each enterprise has specific needs, it is recommended that you discuss your particular situation with your Siebel technical resource.

Planning Backup and Recovery Stages

You may need to recover all Siebel DB2 objects to a prior point in time. Your usual point-in-time recovery techniques may be insufficient in this environment.

In addition to the backup and recovery procedures that are standard for your environment, take a set of DB2 backups at key stages during the upgrade, using your preferred utility. A snapshot of your repositories and environment at these stages protects the progress of your upgrade in the event of a failed subsequent process.

It is recommended that you back up your repository at these key stages of the upgrade:

- Before any upgrade activity is started
 - Before performing unloads
- NOTE:** Unloads must be performed when there is no system activity, so that the database is at a point of consistency.
- After upgrading the Siebel Database schema
 - After the repository merge
 - After upgrading the custom database schema

Review the results of all JCL jobs that you execute during the install or upgrade process. You can use a spool viewer such as IBM's SDSF to inspect the output from these jobs. You can review this information in addition to reviewing the upgrade log files.

CAUTION: When performing a development environment, midtier-centric upgrade, if any of the preschm scripts do not run successfully, data is lost in S_ADDR_PER and cannot be recovered. To prevent this, you should specify that the upgrade script stops when it encounters an error so that you can correct it instead of letting the script run through and then dropping the temp table that contains the data. Use the `-s` option of the `-stvf` command to stop SQL command execution if an SQL error is returned. You should specify the `-s` option when running all Siebel upgrade scripts from the mid-tier.

Creating a Schedule for the Upgrade

Develop a plan for your upgrade based on the objectives and constraints for your deployment.

The following procedures may reduce the time required for your upgrade.

- Run selected processes in advance of the upgrade.

Certain preupgrade tasks can be run at any time prior to the upgrade. These procedures can be performed in advance either for testing purposes or to accommodate down-time constraints.

Examples of procedures that can be performed by a database administrator in advance of your upgrade include [“Preparing the Storage Layout of the Schema” on page 119](#).

- Prepare select processes to run in parallel.

If a large table such as S_EVT_ACT is partitioned, it can run in parallel by transferring shipped statements into the numbered SQL statement.

About Estimating Database Size

Upgrades: All upgrades.

Environments: All environments.

Database upgrade is resource intensive. If the upgrade exceeds available resources, the upgrade halts. You must then resolve resource issues before resuming the upgrade.

To help you estimate the database size required when upgrading to release 7.8, [Table 12](#) shows the number of tables in 4-KB, 16-KB, and 32-KB tablespaces in a sample SIA database in a 7.0.x, 7.5.x and a 7.8 release. [Table 12](#) also shows the space required by the tables.

Because Siebel has adopted a 1:1:1 database schema structure since release 7.7 (one table in each tablespace, one tablespace in each database), these releases require many more 16-KB and 32-KB tablespaces than pre 7.7 releases. However, some tables may not require 16-KB and 32-KB tablespaces if you convert LONG VARCHAR columns to CLOB columns.

NOTE: [Table 12](#) shows the space required by the tables in a release but does not include the space required by indexes.

Actual expected growth may also 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 12. Number of Tables and Space Required in a Sample Siebel DB2 Database by Release

Release...	4-KB Tablespace	16-KB Tablespace	32-KB Tablespace
7.8	3857 (569,442 KB)	459 (237,735 KB)	86 (224,62 KB)
7.5.x	2542 (637,706 KB)	404 (336,384 KB)	22 (143,28 KB)
7.0.x	1995 (398,068 KB)	357 (135,469 KB)	10 (102,39 KB)

Upgrading Your DB2 Software

Upgrades: All upgrades.

Environments: Development environment only.

Before you upgrade, carefully review *Siebel System Requirements and Supported Platforms* on Oracle Technology Network to verify that you are using the currently supported versions of DB2 software.

- **DB2 Version 7 and Version 8.** If you are using an unsupported version of DB2, you must migrate to and thoroughly test the currently supported version before you upgrade to Siebel 7.8. You should perform your DB2 version migration before upgrading your Siebel database. Allow sufficient time to verify the migration before proceeding with the upgrade.
- **DB2 Connect.** Siebel Dedicated Web Clients (Siebel Mobile Web Client in connected mode) and Siebel Servers communicate with DB2 for z/OS through DB2 Connect middleware. Verify that you are using the version of DB2 Connect supported for Siebel 7.8.

NOTE: Make sure that all software meets the requirements specified in *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

About Multilingual Deployments

Upgrades: All upgrades.

Environments: All environments.

The upgrep process imports repository strings and seed data in only the primary language. The primary language, also called the base language, is the language in which data is stored in the Siebel Database.

If you have multilingual deployments, therefore, you must import multilingual repository strings and seed data after performing the upgrep upgrade process. For instructions on how to perform these imports, see 477094.1 (Article ID) on My Oracle Support. This document was formerly published as Siebel Technical Note 447.

See the *Siebel Installation Guide* for the operating system you are using for instructions on installing and configuring multiple Siebel language packs. For a list of supported code pages and encoding strategies, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrading to RC2 or AES Encryption

Upgrades: Releases 7.0.x, 7.5.x.

Environments: All environments.

Release 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 Release 7.0.x 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*.

New Upgrade Features

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: All environments.

Release 7.8

The following new upgrade features were introduced at Release 7.8:

- **Platform support for multiple versions of components.** You can install multiple versions of the Siebel Server, and Siebel Web Server Extension on the same machine. You cannot install multiple versions of the Siebel Gateway Server.

For example, you are upgrading from Release 7.7 to the current release. You have thoroughly tested the Siebel Server in the current release and are ready to install it in your production environment. You can install the new Siebel Server on production machines where Release 7.7 of the Siebel Server is installed. You do not have to take the machines out of service and uninstall the 7.7 Siebel Server before installing the new Siebel Server.

This reduces the downtime required to transition from an installed version of an Enterprise component to a new version.

You cannot run two different versions of the same component on one machine. For example, in the scenario above, you cannot run both the 7.7 version of the Siebel Server and the current release version at the same time.

For information on how to install Enterprise components, including the Gateway Server, Siebel Server, and Siebel Web Server Extension, see *Siebel Installation Guide* for the operating system you are using.

- **Postmerge utilities can be run multiple times.** After the repository merge completes, the postmerge utilities no longer start automatically. You must start them manually, and you can run them multiple times.

If the postmerge utilities do not complete successfully, you do not always have to rerun the repository merge. Depending on the problem, you only have to rerun the postmerge utilities.

Release 7.7

The Incorporate Custom Layouts (ICL) upgrade feature was introduced at Release 7.7. 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 Release 7.7.2 and later. For more information, see [“About the Incorporate Custom Layout \(ICL\) Upgrade Option” on page 264](#).

About User Interface Changes

Upgrades: Releases 7.0.x, 7.5.x.

Environments: All environments.

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

- **Revised navigation scheme.** The key revisions were as follows:
 - The Show menu is replaced by a link bar. When you click on 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, placement of views and controls were based on repository configuration and logic applied at run time. Since Release 7.7, 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.
- **MVG shuttle applets.** MVG shuttle applets were introduced in Release 7.5.x. In Release 7.7, MVGs are shuttle-enabled by default.

Because of these enhancements, you should plan a careful evaluation of the user interface after upgrade. The upgrade logging utility will list any user interface-related problems encountered during upgrade.

For more information on the user interface, see *Fundamentals*.

About Upgrading Access Control

Upgrades: Releases 7.0.x.

Environments: All environments.

Access control was significantly revised in Release 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 Release 7.5, Person, Household, and Service Request can be made visible to multiple organizations, also called Business Units. Release 7.5 introduced several new tables to support this:

- S_CONTACT_BU

- S_ORG_GROUP_BU
- S_SRV_REQ_BU

If you are upgrading from a pre 7.7.x release, the upgrade 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 Release 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.

If you are upgrading from a pre 7.7.x release, the upgrade 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.

The document 476479.1 (Article ID) on My Oracle Support (previously published as Siebel Technical Note 312) provides guidance and best practices for implementing access control. This document 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*.

About the Party Model

Upgrades: Release 6.2.1 only.

Environments: All environments.

Release 7.x introduces 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, Release 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 non-person 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 4 depicts the Party changes to the data model that occur during upgrades from Release 6.2.1 to Release 7.x.

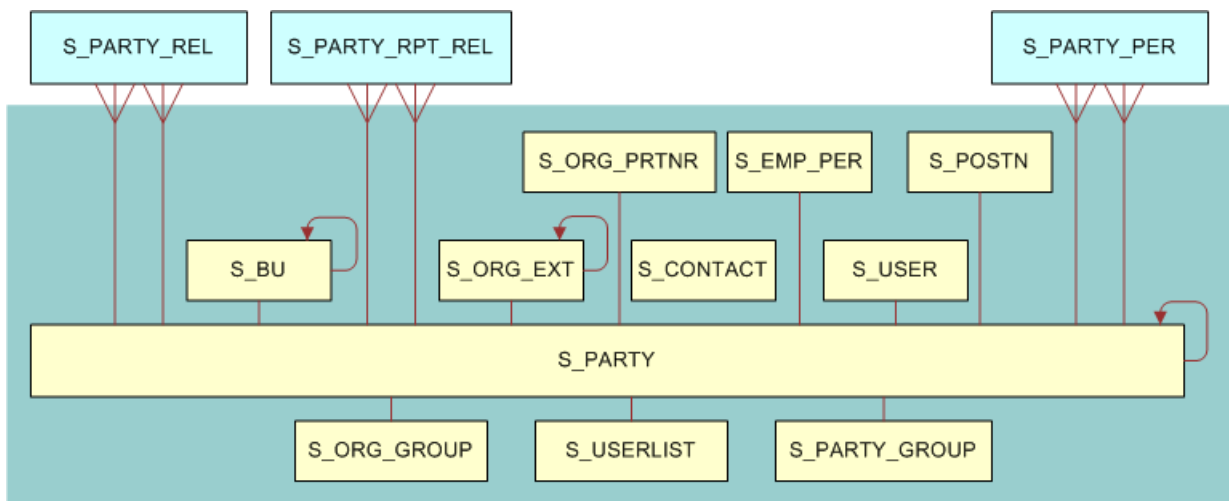


Figure 4. Party Model

How the Party Model Is Implemented During Upgrade

When you upgrade to Release 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 Siebel columns
- Migrates data from S_ORG_INT to S_ORG_EXT, S_BU for standard Siebel 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 would initiate the following actions:
 - Retargets Contact business component to S_PARTY
 - Defines join to S_CONTACT from S_PARTY on the Contact business component
 - Sets implicit join for the Alternate Phone field

Additional Planning Considerations

With Access Control comes two new visibility mode types, catalog visibility and group visibility. For certain standard Siebel Business application configurations, the view mode and picklist mode have catalog visibility and only show data if the catalogs, categories, and access groups are defined.

During upgrade, you must decide whether you will move to the Access Control model, which requires careful planning and preparation of access groups, catalogs, and categories.

If you want to continue using the Multi-Org model, you must modify changes to the default view and picklist configuration as part of the upgrade.

For more details on the data model as it relates to the Party model, see *Siebel Data Model Reference*. For more information on Access Control and Security, see *Security Guide for Siebel Business Applications*.

About the New Column Added to S_SRC_PAYMENT

Upgrades: Release 6.2.1 only.

Environments: All environments.

The S_SRC_PAYMENT table has a new required column, TYPE_CD. The default value for this column is Payment. This value 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.

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

The valid values for this column as shipped with Siebel Business applications are:

- Payment
- Receipt
- Write-Off
- Adjustment

For information on how to update the Lists of Values, see *Configuring Siebel Business Applications*.

About Database Clustered Indexes

Upgrades: Release 6.2.1 only.

Environments: All environments.

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.)

About Migrating HTML Attachments to Base Tables

Upgrades: Releases 7.0.x, 7.5.x.

Environments: All environments.

The upgrade 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.

6

Application Upgrade Planning

This chapter describes the changes between Siebel 7.8 applications and previous releases that can affect the upgrade process. It includes the following topics:

- “Upgrade Planning for Address Data Migration”
- “Upgrade Planning for Siebel Employee Relationship Management (ERM)” on page 104
- “Upgrade Planning for Siebel Marketing” on page 105
- “Upgrade Planning for Siebel Workflow Designer” on page 110
- “Upgrade Planning for Handheld Devices” on page 111
- “Upgrade Planning for Resonate Central Dispatch” on page 112
- “Upgrade Planning for String Translation” on page 112

Upgrade Planning for Address Data Migration

Upgrades:

- From Release 6.2.1 of Siebel Financial Services applications on IBM z/OS platforms to Release 7.8.x of Siebel Industry applications on IBM z/OS platforms
- From Release 7.0.x of Siebel Financial Services applications to Release 7.8.x of Siebel Industry applications
- From Release 7.8.x of Siebel Business applications to Release 7.8.x of Siebel Industry applications

NOTE: This topic does not apply to Siebel Business applications (HOR) that you are upgrading to a later release of Siebel Business applications (HOR).

Release 7.8 changes the way address information is stored. You must perform several tasks to make sure 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.

Release 7.8

In Release 7.8, 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.
- 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.
- Uniqueness of addresses in S_ADDR_PER is enforced only on ADDR_NAME.

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 three 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. During the database upprep, you do this after running the Database Server Configuration utility but before running the Siebel 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 [“Preparing Address Data for Upgrade” on page 143](#). 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: Releases 7.0.x, 7.5.x.

Release 7.7 of ERM introduced important changes to workflows, Siebel Training, Microsite management, and Group News.

Workflows

Release 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. Release 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: Releases 7.0.x, 7.5.x.

Release 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. Release 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 5.

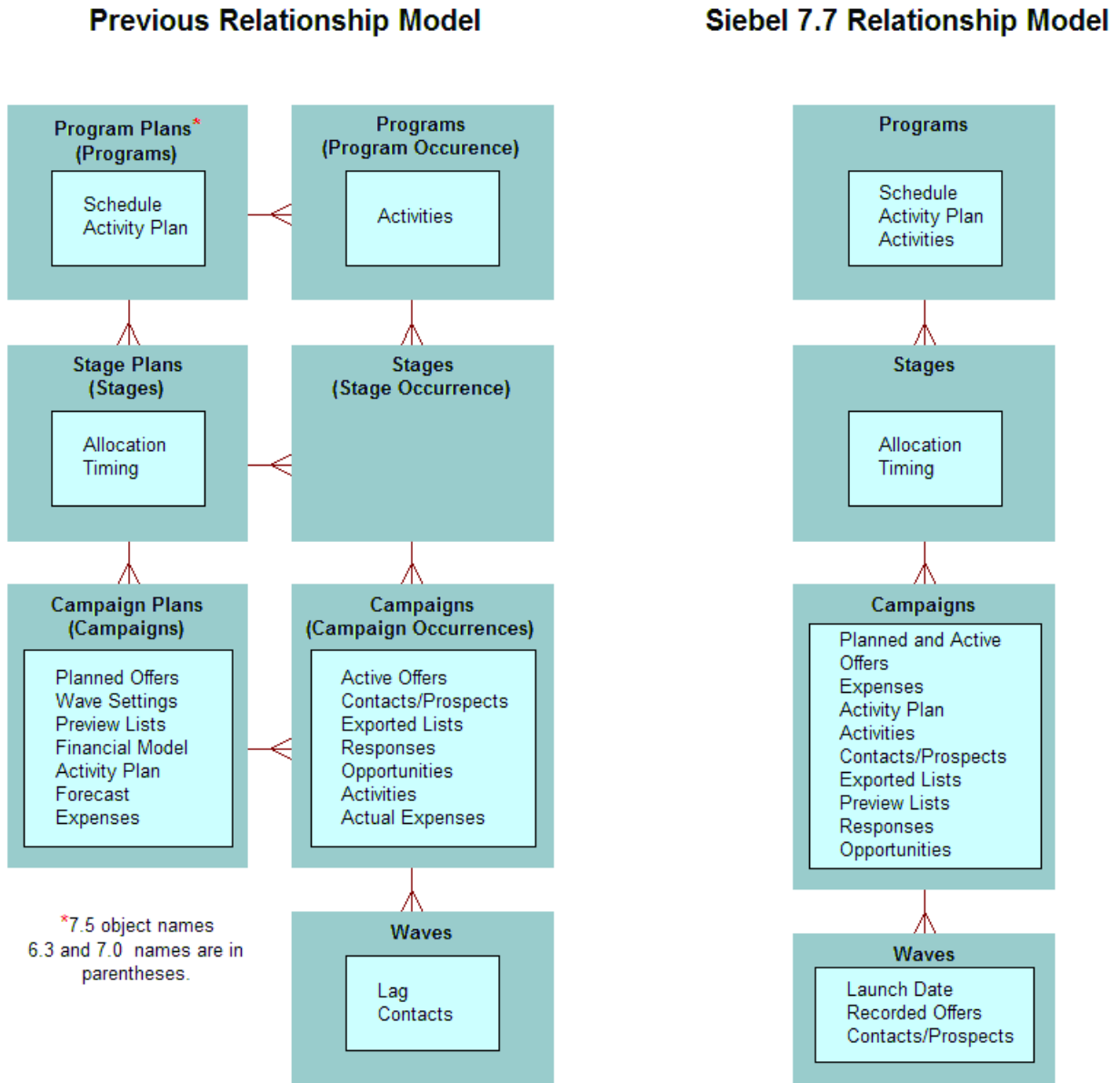


Figure 5. Relationship Model for Siebel Marketing

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 Enterprise 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 (SObjMgr), eMarketing AOM (eMarketObjMgr), or the eEvents AOM (eEventsObjMgr). They are actively used in the application.

[Appendix B, “Siebel Marketing Upgrade Reference”](#) lists the following obsolete items:

- Server components
- Business objects and business components
- Screens and views
- Data that is obsolete

About Siebel Marketing Customizations

Release 7.7 introduced the Incorporate Custom Layouts (ICL) feature. 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 Siebel Marketing Database Changes

Table 13 summarizes Siebel Database changes that apply to Siebel Marketing.

Table 13. Summary of Siebel Database Changes at 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	Updates Status values (STATUS_CD) for campaigns and programs. Updates campaign plans and stand-alone campaigns to use the same CAMP_TYPE_CD. Updates the values for the S_SRC. RESPNSE_TYPE.
Inserts into S_DD_CAMP_WAVE and S_CAMP_LD_WAVE	Creates a Load record for each campaign occurrence. Load numbers are assigned based on date of execution. Creates one load wave for each stand-alone campaign. Sets the status of the load wave to the corresponding execution status from the original campaign occurrence.
Inserts into S_CAMP_WAVE_DCP	Records the offer history for load wave records (preupgrade occurrences).
Index change and update to S_CAMP_CON	Modifies the user key to use Campaign ID + Contact ID + Load Number + Token Number. When campaign occurrences are converted to Load records, S_CAMP_CON is updated to the appropriate Load number for the campaign contact.
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	Re-parents activities from campaign occurrences to campaigns. Re-parents activities from program occurrences to programs.
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.

Table 13. Summary of Siebel Database Changes at Siebel 7.7

Upgrade Step	Description
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 14 describes how Siebel Marketing data will be migrated.

Data that is obsolete as of Release 7.8 is listed in [Appendix B, "Siebel Marketing Upgrade Reference."](#)

Table 14. Siebel Marketing Data Migration at 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	

Table 14. Siebel Marketing Data Migration at Siebel 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
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 Release 7.7 Segment Designer.

Upgrade Planning for Siebel Workflow Designer

Upgrades: Releases 7.0.x, 7.5.x.

In Release 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 production environments, make sure that they include all of the necessary workflows.

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.

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 S_WFA.
- Repository tables containing workflow information begin S_WFR.
- Siebel Database tables that contain workflow information for releases prior to Release 7.7 begin S_WF_ (note the underscore after WF). After the upgrade to Release 7.7, these tables are obsolete and are not referenced by applications.

Upgrade Planning for Handheld Devices

Upgrades: Releases 7.0.x, 7.5.x.

Device operating system support, third-party product support, and application configuration management for handheld devices were revised in Release 7.7. Before the upgrade, verify that handheld devices are running an OS supported by Release 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 Release 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 *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Active Sync Support

The third-party product Active Sync must be at version 3.7.1 or higher as of Release 7.7. For a complete description of supported third-party products, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Handheld Device Administration

Release 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 Release 7.8.

Upgrade Planning for Resonate Central Dispatch

Upgrades: Releases 7.0.x, 7.5.x.

Support for Resonate Central Dispatch is discontinued as of Release 7.7. It has been replaced by a load balancing module that is included in the Siebel Web Server Extension. In addition, Siebel Systems has certified several third-party HTTP load balancers for use with the Siebel Web Server Extension.

For a description of the Siebel load balancing module, see *Siebel System Administration Guide* and *Siebel Installation Guide* for the operating system you are using. For a list of supported HTTP load balancers, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

Upgrade Planning for String Translation

Upgrades: Releases 7.0.x, 7.5.x.

Release 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 Siebel assistance before upgrading. If you do not obtain assistance, you could lose or corrupt data during upgrade.

7

Basic Database Preparations

This chapter describes the database tasks you should complete before upgrading to release 7.8. It includes the following topics:

- [“Preparing Tables and Views for Upgrade”](#)
- [“Preparing Custom Indexes for Upgrade” on page 116](#)
- [“Exporting Interface Table Data” on page 117](#)
- [“Preserving Dock Objects and Visibility Rules” on page 117](#)
- [“Verifying Database Server Configuration” on page 117](#)
- [“Securing AIX Memory Allocation Segment Space” on page 118](#)
- [“Creating Storage Groups” on page 118](#)
- [“Preparing the Storage Layout of the Schema” on page 119](#)
- [“Reviewing Tablespace Size” on page 129](#)
- [“Backing Up the Database” on page 131](#)
- [“Adding a Siebel User” on page 132](#)

Preparing Tables and Views for Upgrade

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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.

Release 7.8 does not support customized triggers. If you have created customized triggers on your Siebel base tables, disable them before you perform the upgrade. You must recreate them after the upgrade is finished.

- 3 Drop customized views on Siebel tables.

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

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 Custom Indexes for Upgrade

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 Release 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*.

Dropping Custom Temporary Tables

Release 7.x does not support custom tables created at the database level. If you have temporary tables or non-Siebel tables, drop them before you perform the upgrade. If the upgrade process detects any column with a data type under the table that is not acceptable by Siebel, the upgrade fails.

Considerations for Clustered Indexes

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

Exporting Interface Table Data

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 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.

Preserving Dock Objects and Visibility Rules

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Changes to visibility rules and dock objects require the assistance of Technical Support or Siebel Expert Services.

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.

Verifying Database Server Configuration

Upgrades: All upgrades.

Environments: Development environment only.

Verify that your development Siebel Database Server configuration meets or exceeds Siebel requirements as described in *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

- Informational APAR pq83289 is required for loading multiple tables to the same database in parallel.
- The upgrade sets the TRACKMOD parameter the default value for tablespace objects: YES.

- The DSNZPARM CMTSTAT should be set to INACTIVE to prevent timeout errors from occurring. If you do not run INACTIVE, then set IDTHTOIN to 0 (Inactive).

CAUTION: It is very important that the required maintenance level listed in informational APAR ii13012 is fully applied, and that all DSNZPARMs required for Siebel implementation on DB2 v7 or v8 are set up correctly.

Before you begin your upgrade, verify your database configuration. The consequence of exceeding available resources is a halted upgrade that requires you to allocate time to adjust the environment and then resume the upgrade.

When you upgrade from a previous version of Siebel Business applications to Release 7.8, expect database growth in the range of 30-50%. The amount by which your database grows may vary widely, depending on your database configuration, row size of tables, and data content.

The growth percentage will increase depending on how you size your database and configure default storage for database tablespaces. For example, if you set the default storage for your initial or next extent in a given DB2 tablespace to 10 KB, that tablespace will grow by a smaller percentage than if you set it to 100 KB.

Securing AIX Memory Allocation Segment Space

Upgrades: All upgrades.

Environments: All environments.

Platforms: UNIX only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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.

Creating Storage Groups

Upgrades: All upgrades.

Environments: Development environment only.

Before upgrading the Siebel Database Server, your DBA must create storage groups on the target database. The upgrade process assumes that the names for the storage groups in the target database are the same as those defined in the source database. For this reason, the names used to define the storage groups for the target database should be the same as those defined on the source database. If the names are not the same, your DBA must manually modify the schema.sql file so that the names match.

Preparing the Storage Layout of the Schema

Upgrades: All upgrades.

Environments: Development environment only.

Before starting the 7.8 upgrade, you must prepare the storage control file you will use during the upgrade. The DBA can prepare the storage control file at any time before the upgrade.

The storage control file contains storage information, including bufferpools and storage groups, that is used as the basis for the storage layout of your new 7.8 Siebel database. Even if you are using a preconfigured storage layout, you must make sure that the layout is valid for your schema.

NOTE: Release 7.7 includes a new schema structure. For more information on this structure, see “Planning Your Upgrade” on page 87.

Preparing the storage control file consists of four procedures:

- “Modifying the Storage Control File” on page 120
- “Extracting the Storage Control File” on page 122
- “Validating the Extracted Storage Control File” on page 126
- “Reviewing the Storage Control File” on page 127

NOTE: You must validate the storage control file after you extract it and after you modify it.

There are different starting points from which you can customize your storage layout:

- **Scenario 1.** Begin with a Siebel-provided storage layout template, import the template into the Siebel Database Storage Configurator (dbconf.xls), customize it, then export it as your customized layout.
- **Scenario 2.** Use your current configuration from an existing database layout and merge it with one of the Siebel-provided templates. This can then be imported to Siebel Database Storage Configurator for further manipulation.

NOTE: For more information on the Siebel Database Storage Configurator, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

To prepare the storage control file, use the Database Server Configuration Wizard to extract the storage layout of your source database from the DB2 catalog. As part of the extraction process, you can merge the storage layout information from your existing database with information you already input into a storage template file or information in a template provided by Siebel. As an alternative, instead of extracting the storage layout of your existing database, you can use a Siebel-provided template as the storage control file.

Siebel Systems provides three templates for the storage control file:

- **storage_np.ctl.** Contains no partitioning scheme
- **storage_p.ctl.** Contains partitioning scheme for a set of tables on an ASCII database
- **storage_p_e.ctl.** Contains partitioning scheme for a set of tables on an EBCDIC database

For more information about templates for the storage control file, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

CAUTION: The storage layout for partitioned tables ships with a default code page of ASCII. If you are upgrading an EBCDIC database, you must use the EBCDIC storage control file `storage_p_e.ctl`.

Modifying the Storage Control File

There are three methods by which you can configure storage space:

- **Method 1.** This method consists of performing a standard Siebel database installation by running the Database Server Configuration Wizard, choosing the Generate DDL into a File installation option, and specifying as input one of the Siebel storage control file templates. This process generates the following:
 - A storage control file, based on the Siebel template file you selected, that incorporates the configuration information you entered when you ran the Database Server Configuration Wizard. This file is generated in the `dbsrvr\DB2390` (Windows) or `dbsrvr/DB2390` (UNIX) directory
 - A `schema.sql` file that is applied on the z/OS host to create the Siebel schema. The `schema.sql` file is based on the customized storage control file generated by the database install

Using these files, you can then configure storage space in any of the following ways:

- Amend the storage control file generated during the database install in the `dbsrvr\DB2390` (Windows) or `dbsrvr/DB2390` (UNIX) directory (see also Method 2).
- Apply the `schema.sql` file generated by the database install on the DB2 host to create the Siebel schema, then amend the schema using native DB2 tools. Extract the storage control file from the DB2 catalog; the file will include the changes you have made.

NOTE: You access the extract utility Siebel provides through the Database Server Configuration Wizard. This utility allows you to extract information from the DB2 catalog. You can use this extract utility any time you want to create a new storage control file, based on the DB2 catalog.

- Amend the `schema.sql` file generated by the database install directly, apply it on the DB2 host to create the schema, and then extract the storage control file, which will include the changes you made.

For information on installing the Siebel database and extracting storage control files, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

- **Method 2.** This method consists of manipulating the storage control file (`storage.ctl`) directly by opening it with a text editor program. This method can be used if you understand the file structure.

You can use this method to amend one of the Siebel-supplied storage control files or to amend a storage control file that you have extracted from another Siebel schema.

- **Method 3.** This method consists of using the Siebel Database Storage Configurator tool. The Siebel Database Storage Configurator tool is a Microsoft Excel macro (`dbconf.xls`) that is installed in the `dbrvr\db2390` (Windows) or `dbrvr/db2390` (UNIX) subdirectory of your installation directory. This tool allows you to import a storage control file, amend it, validate the syntax of your changes, and then save it.

UNIX customers must transfer `dbconf.xls` and the `.ctl` files to their Microsoft Windows environment. Use BINARY FTP transfer for the `dbconf.xls` file.

For information on using the Siebel Database Storage Configurator, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

NOTE: Validate the storage control file after you modify it. See [“Validating the Extracted Storage Control File”](#) on page 126.

About Extracting the Storage Control File

The information in your storage control file comes from the source database. When you extract the storage control file, you can choose one of two methods:

- **Extract from Catalog.** This method extracts the storage layout of your source database from the DB2 catalog as-is. The output is a new storage layout.

NOTE: This method can also be used to perfect your target database layout by performing a *dummy* installation, manipulating the schema layout through native utilities, then extracting the customized layout.

- **Extract from Catalog and Merge with Template.** This method preserves your existing layout. This merges storage layout information from your source database with information that you already entered into a storage template file provided by Siebel. This preserves your prior layout, and your output is the prior physical layout merged with a 7.x storage layout template.

The first time that you run an upgrade, when you extract your storage control file, you must use the Extract from catalog and merge with template option to preserve your existing layout. The list below details scenarios that may occur during the extract and merge process and the behavior you can expect in these cases:

- **A database object exists in the existing database but not in the template.** The definition of the database object is output to the new storage control file.

- **A database object is specified only in the template file.** The definition of the database object is output to the new storage control file.
- **A database object is specified in both the existing database and the template.** The layout of the existing database is extracted as the default. However, you can manually override this behavior by creating a file called `overri de. i np` in the `bi n` directory under `SI EBSRVR_ROOT`. Place any tables that you want to override into this file.

CAUTION: Partitioned tables were not supported for the 6.x releases. If you applied partitioning to release 6.x tables, carefully review the extracted data and modify it as appropriate. If you have questions, contact Siebel Technical Support or Professional Services.

For additional information on the Extract option, see *Implementing Siebel Business Applications on DB2 UDB for z/OS* and [“Extracting the Storage Control File.”](#)

Extracting the Storage Control File

To extract the storage control file from the source database, follow the procedure outlined in this topic.

To extract the storage control file

- 1 Run the Database Server Configuration Wizard.

For information on running the Database Server Configuration Wizard, see [“Running the Database Server Configuration Utility Under Windows”](#) on page 168 or [“Running the Database Server Configuration Utility Under UNIX”](#) on page 169.

- 2 Enter the information shown in [Table 15](#) when prompted by the Database Server Configuration utility. Collect this information and verify it before running the utility.

Table 15. Information Required for the Database Server Configuration Utility—Extract and Merge Option

Screen Name	Required Information
Siebel Gateway Name Server Address	Windows only. The Siebel Gateway Name Server machine name and the Enterprise Server name.
Siebel Server Directory	The absolute path of the directory where the Siebel Server is installed—for example <code>D:\sea_7x\si ebsrvr</code> . For UNIX, do <i>not</i> enter the string <code>\$SIEBEL_ROOT</code> .
Siebel Database Server Directory	The absolute path of the directory where the Siebel Database Server is installed—for example <code>D:\sea_7x\dbsrvr</code> .
RDBMS Platform	Choose IBM DB2 UDB for z/OS.
Siebel Database Operation	Choose Run Database Utilities.
Database Utilities Options	Choose Configure Database

Table 15. Information Required for the Database Server Configuration Utility—Extract and Merge Option

Screen Name	Required Information
Database Configuration Options	Choose Extract Storage File to extract a storage control file.
Extract Options	<p>Extract from Catalog. This option extracts the storage layout of your source database from the DB2 catalog as-is. The output is a new storage layout.</p> <p>Extract from Catalog and Merge with Template. This option preserves your existing layout. This option merges storage layout information from your source database with information that you already entered into a storage control file, only taking objects from the template that do not already exist in the catalog.</p> <p>NOTE: The first time that you run an upgrade, you must use the Extract from Catalog and Merge with Template option, thereby preserving your existing layout.</p>
Source Database ODBC Data Source Name	<p>Verify the ODBC name for connecting to the source Siebel Database for your upgrade. If it is not correct, enter the correct ODBC name.</p> <p>The ODBC data source <i>must</i> have the same name as the subsystem. When you set up the ODBC connection in DB2 Connect, use the actual subsystem name for the database alias.</p> <p>Windows: To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DNS tab to find the name of your ODBC data source.</p> <p>UNIX: To find the name of your ODBC data source, type: vi \$ODBCINI.</p>
Source Database User Name	<p>Enter the source user name and password for the Siebel administrator of the source database for your upgrade.</p> <p>NOTE: The source database user name (user ID) needs to have authorization to set CURRENT SQLID.</p>
Source Siebel Schema Qualifier and Security Group ID.	<p>Source Schema Qualifier. Enter the eight-character identifier that designates the Siebel Schema for your source database. This is also an authorization ID. The schema qualifier must start with a letter, cannot contain special characters, and must be entered in uppercase.</p> <p>Source Security Group ID. Enter the user ID of the group to whom schema access is granted, for example, SSEROLE.</p>
Code Page Encoding Scheme	Indicate whether your DB2 subsystem is ASCII or EBCDIC.

Table 15. Information Required for the Database Server Configuration Utility—Extract and Merge Option

Screen Name	Required Information
Siebel Schema Layout	<p>Choose Siebel Schema without Partitioning if you want all tables only in segmented tablespaces.</p> <p>Choose Siebel Schema with Partitioning if you want a layout that includes a set of tables that is recommended for partitioning. The remaining nonpartitioned tables are in segmented tablespaces.</p>
Default Tablespace	Enter the name of the default tablespace.
Storage Group for Tablespaces Storage Group for Indexes	<p>Indicate the values for the following parameters:</p> <p>Storage Group for Tablespaces. Enter the name of the table storage group.</p> <p>Storage Group for Indexes. Enter the name of the index storage group.</p>
4KB Buffer Pool Name 16KB Buffer Pool Name	<p>Indicate the values for the following parameters:</p> <p>4KB Buffer Pool Name. Enter the 4-KB buffer pool name for your tablespaces or accept the default name, BP1. This buffer pool should already be activated and have access to it granted by the DBA.</p> <p>16KB Buffer Pool Name. Enter the 16-KB buffer pool name for your tablespaces or accept the default name, BP16K1. This buffer pool should already be activated and have access to it granted by the DBA.</p>
32KB Buffer Pool Name Index Buffer Pool Name	<p>Indicate the values for the following parameters:</p> <p>32KB Buffer Pool Name. Enter the 32-KB buffer pool name for your tablespaces or accept the default name, BP32K1. This buffer pool should already be activated and have access to it granted by the DBA.</p> <p>Index Buffer Pool Name. Enter the buffer pool name for indexes or accept the default name, BP2. This buffer pool should already be activated and have access to it granted by the DBA.</p>
Database Name Prefix	<p>Enter the prefix for your target database name. The default prefix is SIDB.</p> <p>This prefix consists of up to four of the first characters in the names of your logical Siebel Databases. This prefix must start with a letter and cannot contain any special characters. All database names end in numbers.</p>

Table 15. Information Required for the Database Server Configuration Utility—Extract and Merge Option

Screen Name	Required Information
Storage Control File	<p>Enter the directory path and name for the storage control file created by this process:</p> <ul style="list-style-type: none"> ■ Windows: Accept the default value displayed in the Storage Control File field (this is the <i>DBSRVR_ROOT\DB2390</i> directory, for example, <i>D:\DBSRVR_ROOT\DB2390\storage.ct1</i>) or use the Browse button to navigate to a different directory. ■ UNIX: Enter the path to your storage control file.
Log Output Directory	<p>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, slashes, or symbols in the name of the log output.</p> <p>CAUTION: If you are restarting or recovering a previous incomplete upgrade, do not change the Log Output Directory that you previously selected.</p>

- 3 Save the configuration information you have entered and launch the Siebel Upgrade Wizard as described in the following topics:
 - [“Running the Database Server Configuration Utility Under Windows” on page 168.](#)
 - [“Running the Database Server Configuration Utility Under UNIX” on page 169.](#)
- 4 When the Siebel Upgrade Wizard appears, click OK. The Upgrade Wizard connects to the z/OS host, extracts the storage control file and copies it to the file that you specified (see [Table 15 on page 122](#)).

If you selected the Extract from Catalog and merge with Template option, this process reads the database catalog and merges your prior custom database layout with one of the 7.8 Siebel Database layout templates (located in the *DBSRVR_ROOT\db2390* directory). New objects take a layout from one of the layout templates.

NOTE: If you choose a Siebel storage control file template that includes partitioning, and the existing database schema does not include partitioning, by default, the existing database objects are not partitioned in the storage control file generated (that is, the database catalog overrides the templates) unless you specify the table names in the override input file (*override.inp*).

By default, new tables are created as one-table-per-database to prevent concurrency and locking errors.

Once the merged storage control file is created, you must verify it against either the source or target database. See [“Validating the Extracted Storage Control File” on page 126](#) for information on this task.

Validating the Extracted Storage Control File

When you have extracted your existing database storage control file and merged it with a Siebel 7.8 template in preparation for the 7.8 upgrade, you must validate the storage control file. (You must also validate the storage control file any time you modify it.) The validation process validates that the tables are the right length for the target schema. Do not proceed with the Siebel 7.8 upgrade until the validation process runs without error.

Be aware of the following:

- For development environment, midtier-centric upgrades, the validation process validates the storage control file for the source database, so you must use the source tableowner.
- For production environment upgrades, the validation process validates the storage control file for the target database, so you must use the target tableowner.

The validation produces two output files in the `log` directory under `SI EBSRVR_ROOT`.

- `db_config_si_ebel.log`
- `db_config_validation.log`

NOTE: When validating the extracted storage control file, the validation job calls for the `schema.ddl` file as an example. Make a copy of `ddl.ctl` and rename it `schema.ddl`, keeping the copy in the same location as `ddl.ctl`.

If you are validating a storage control file for an EBCDIC database, the validation process does not report all errors the first time it is run. You must rerun the validation process after fixing all reported errors. If new errors are reported on a subsequent validation, you must fix the errors and run the validation again. You may proceed after the validation reports no errors.

NOTE: It is recommended that you enter all values in uppercase letters.

To validate the storage control file

- 1 Launch the Database Server Configuration Wizard as described in [“Extracting the Storage Control File” on page 122](#).

In the following steps, for Windows users, screen refers to a standard, application window. For UNIX users, the Siebel Upgrade Wizard displays in a shell. Screen refers to the pages of text that display in the shell.

- 2 Enter the information requested on each screen and click or press the appropriate button for your operating system to proceed:
 - Windows: Click Next.
 - UNIX: Press Enter.
- 3 Specify the following values:
 - a On the Database Configuration Options screen, select the Validate Storage Control File option.
 - b On the Schema File screen, specify the following values:

- **Schema File.** Specify the directory path and filename of the file against which the extracted file is to be validated. For development environment upgrades, specify the *ddl.ctl* file. For production environment upgrades, specify the *schema.ddl* file.
 - **Storage Control File.** The name of the storage control file that you want to validate (this is the file that you extracted in “[Extracting the Storage Control File](#)” on page 122).
- 4 On the Configuration Parameter Review screen, review the configuration values you entered on the previous Configuration utility screens.

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

If you need to change any values, do the following:

- Windows: Use the Previous and Next buttons to access the desired screens, then to return to this screen. When you are satisfied, click Finish to launch the Siebel Upgrade Wizard.
 - UNIX: Enter N, and rerun the Database Server Configuration Wizard by entering `dbsrvr_config.ksh`. This allows you to reconfigure with different values. When you are satisfied, enter Y to launch the Siebel Upgrade Wizard (`srvrupgwi.z`).
- 5 When the Upgrade Wizard appears, click OK start the storage control file validation process.

NOTE: If the Validate Storage Control File process fails because tablespaces for your Siebel schema are not large enough to hold the new table definitions, examine the `dbvalidate_report.log` file for further information. Look for the bufferpools for any offending tablespace, and increase them as necessary in the `storage.ctl` file.

- 6 Review the log files, `dbvalidate_report.log` and `dbvalidate_schema.log`, which are generated in the `$SIEBEL_ROOT\LOG\dbconfi_g_validate_mf\output` directory (Windows) or the `$SIEBEL_ROOT\LOG/dbconfi_g_validate_mf/output` directory (UNIX).

Reviewing the Storage Control File

After you have extracted the storage control file, you must review it. Your storage control file is located in the directory that you specified when you ran the Database Server Configuration Wizard to extract the file. Check the following parameters in the control file and modify them as appropriate for your database needs. Once you have a storage layout that you are satisfied with, you are ready to continue with your upgrade.

NOTE: You must carefully review and edit the storage control file to meet your needs. Every time you modify the file, you must validate it again. See “[Validating the Extracted Storage Control File](#)” on page 126.

Do not change the defaults for the following parameters in [Object 1] in the storage control file:

- IndexStogroup
- IndexBp
- PriQty
- SecQty

[Object 1]

Type = Defaults
Name = Defaults
Database = \$DbnamePrefix0000
Tablespace = SI EBTS00
Stogroup = \$StogroupTables
IndexStogroup = \$StogroupIndexes
IndexBp = \$IndexBufferPool
Bufferpool = \$4KBufferPool
Locksize = Page
SegSize = 32
LockMax = 0
Priority = 7200
SecQty = 14400
PctFree = 17
FreePage = 0
Compress = No
Define = No
Erase = No
CCSID = \$DbType

[Object 2]

Type = Database
Name = \$DbnamePrefix0000
LockSize = Page

[Object 3]

Type = Tablespace
Name = SI EBTS00
Database = \$DbnamePrefix0000
Bufferpool = \$4KBufferPool
Stogroup = \$StogroupTables

LockSize = PAGE
 LockMax = 0
 SegSize = 32
 PriQty = 7200
 SecQty = 14400
 PctFree = 17
 FreePage = 0
 Compress = No
 Partitions = 0
 Define = 0
 Erase = 0

Reviewing Tablespace Size

Upgrades: All upgrades.

Environments: Development environment only.

Certain tables require larger tablespaces in Release 7.8 to accommodate additional columns. Review the tables listed in [Step 16 on page 129](#) and verify that the page size is sufficient to accommodate new columns (some of these tables may not have existed in your schema in previous releases.)

NOTE: If the S_EMPLOYEE table existed in your schema during previous releases, you must move S_EMPLOYEE to a 16-KB bufferpool before performing an upgrade because an additional column is added to this table by the upgrade. S_EMPLOYEE has no default tablespace name, because it is obsolete.

Specify an entry in the storage control file for each obsolete table and extended table that you want to move to the 1:1:1 model. The 1:1:1 model was new in Release 7.7. For more information, see [“Planning Your Upgrade” on page 87](#). It is recommended that you validate the storage control file, particularly where you have schema changes.

Drop all views (including custom views) on a table before you move it.

Table 16. Tables with Additional Columns

Table Name	Default Tablespace Name	New Page Size
S_ASSET	H0477000	16 KB
S_COLUMN	H0658000	16 KB
S_EMPLOYEE	H0997000	16 KB
S_CRSE_URL	H0738000	16 KB

Table 16. Tables with Additional Columns

Table Name	Default Tablespace Name	New Page Size
S_DD_TRGT_GRP	H0875000	16 KB
S_I C_CALC_I T	H1158000	16 KB
S_PART_RPR	H1523000	16 KB
S_PERF_MGR_APPR	H1534000	16 KB
S_PROD_DEFECT	H1626000	16 KB
S_PROD_I NT	H1639000	16 KB
S_SCHMST_DBSCPT	H1808000	32 KB
S_SRC_PAYMENT	H1894000	16 KB
S_SRV_REQ1_FNX	F2689000	16 KB
S_SRV_REQ1_FNXM	F2690000	16 KB
S_TABLE	H1966000	16 KB
S_UPG_KI TWI Z	H2009000	16 KB

Table 17 lists the EIM tables that need to be evaluated for sufficient tablespace size.

Table 17. EIM Tables with Additional Columns

Table Name	Default Tablespace Name	New Page Size
EIM_ACCOUNT	H0004000	32 KB
EIM_ADDR_PER	H0019000	16 KB
EIM_AGREE_DTL	H0022000	16 KB
EIM_AGREEMENT1	H0021000	32 KB
EIM_ASSET_DTL	H0033000	16 KB
EIM_BASELN_DTL	H0041000	16 KB
EIM_CALL_LST	H0046000	32 KB
EIM_CONSUM_DTL	H0064000	16 KB
EIM_CONTACT	H0065000	32 KB
EIM_CONTACT1	H0066000	32 KB
EIM_COURSE	H0073000	32 KB
EIM_ENTLTMPL	H0121000	16 KB
EIM_FN_ASGNGRP	F2300000	16 KB
EIM_FN_PAYEE1	F2346000	16 KB

Table 17. EIM Tables with Additional Columns

Table Name	Default Tablespace Name	New Page Size
EIM_FN_PAYEE2	F2347000	16 KB
EIM_FN_REVN1	F2357000	32 KB
EIM_GROUP	H0147000	16 KB
EIM_IC_INCENTV	H0153000	16 KB
EIM_MDF_DTL	H0181000	16 KB
EIM_ORDER	H0198000	32 KB
EIM_ORDER1	H0199000	16 KB
EIM_ORDER_ITEM	H0201000	32 KB
EIM_ORDER_SHIP	H0203000	16 KB
EIM_PAYMENT	H0215000	32 KB
EIM_PAYMNT_DTL	H0216000	16 KB
EIM_PDSHIP_DTL	H0219000	16 KB
EIM_PL_ITEM_DTL	H0228000	16 KB
EIM_PRODINVLOC	H0249000	16 KB
EIM_PRSP_CON	H0271000	32 KB
EIM_PT_RPR_DTL	H0276000	16 KB
EIM_QUO_ITEM_DTL	H0286000	16 KB
EIM_QUOTE1	H0281000	16 KB
EIM_QUOTE_ITEM	H0283000	32 KB
EIM_SRC1	H0301000	16 KB
EIM_SRC2	H0302000	16 KB
EIM_SRC_DBM	H0303000	16 KB
EIM_SRC_EVT	H0305000	32 KB
EIM_TARGET_DTL	H0317000	16 KB

Backing Up the Database

Upgrades: All upgrades.

Environments: Development environment only.

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

It is a recommended practice that you back up your database at key stages of the upgrade:

- Before any upgrade activity is started
- After upgrading the Siebel Database Schema or Custom Database Schema (upgrep+upgphys)
- After the repository merge

Adding a Siebel User

Upgrades: All upgrades.

Environments: Development environment only.

Through Siebel Client application administration, add the Siebel user (with administrative privileges) who executes the Database Server Configuration Wizard. This is the person whose user ID is entered when the Database Server Configuration Wizard prompts for Database User Name. For further information on adding Siebel users, see *Applications Administration Guide*.

8

Preparing Application Data for Upgrade

This chapter covers the tasks involved in preparing Siebel application data for upgrade. It includes the following topics:

- “Preparing Workflow Processes for Upgrade”
- “Identifying Seed Data Customizations” on page 134
- “Preparing Mobile User Data for the Database Upgrade” on page 140
- “Setting Up Campaign Status Values for Siebel Marketing Upgrade” on page 141
- “Preparing Address Data for Upgrade” on page 143
- “Preparing Products for Upgrade” on page 144
- “Preserving Marketing Segment Descriptions” on page 144
- “Identifying and Resolving Duplicate Row IDs” on page 145
- “Preparing Siebel eChannel Data for Upgrade” on page 146
- “Preparing Forecasting Data for Upgrade” on page 146
- “Preparing Products and Quotes for Upgrade” on page 147
- “Verifying Household Data Integrity” on page 147
- “Preparing Financial Services Application Tables for Upgrade” on page 149
- “Preparing Siebel Configurator Data for Upgrade” on page 149
- “Setting the Value of S_SRC_PAYMENT.TYPE_CD” on page 150

Preparing Workflow Processes for Upgrade

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Deactivate all workflow processes before doing the upgrade. This ensures that run-time event actions associated with old workflows are cleaned up properly.

Development Environment Upgrades

Before upgrading your development environment, make sure that all customized workflows and all seed workflows from the production database are present in the development environment Siebel Database. The upgrade will move them to the Siebel Repository and also update the seed workflows. The upgrade migrates custom workflows from S_WF_* tables to S_WFR_* tables in the Siebel Repository.

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.

Siebel Systems provides a script and an input file for migrating workflow data between your production database and your development database. For information, see 477668.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Alert 1016.

Identifying Seed Data Customizations

Upgrades: Release 7.8.1.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The upgrade from Release 7.8.1 to 7.8.2 deletes existing seed data for customer and order management applications and inserts new seed data. This topic describes how to run a report that identifies your customizations to Release 7.8.1 seed data for these applications.

After the upgrade, use this report to apply these customizations to Release 7.8.2 seed data as desired.

Perform the following tasks in the order shown to identify seed data customizations:

- Creating the report view
- Running the report
- Reviewing the report
- Dropping the report view

Prerequisites: You must have Microsoft Excel, and you must be able to enter a stand-alone ^ (circumflex) from the keyboard. To enter a circumflex if it is not on your keyboard, press and hold the left Alt key; then enter 94 on the numeric keypad, and release the Alt key.

Creating the Report View

A script for creating the report view is included in Release 7.8.2.

To create the report view

- 1 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

2 Type the following command:

■ Windows:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_create_view.txt /h /separator /
/L LogFileLocation\rpt1_create_view.log
ScriptLocation\create_view_vod_ver_view.sql
```

■ UNIX:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation/rpt1_create_view.txt /h /separator /
/L LogFileLocation/rpt1_create_view.log
ScriptLocation/create_view_vod_ver_view.sql
```

■ z/OS:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation/rpt1_create_view.txt /q SchemaQualifier /h /separator /
/L LogFileLocation/rpt1_create_view.log
ScriptLocation/create_view_vod_ver_view.sql
```

where:

Tableowner = Siebel Database tableowner account name

Password = Tableowner password

ODBCDataSource = Data source name for the Siebel Database

OutputFileLocation = Location of the output file:

□ Windows: *SIEBEL_ROOT\log\rpt1_create_view.txt*

□ UNIX: *\$SIEBEL_ROOT/log/rpt1_create_view.txt*

SchemaQualifier = DB2 schema qualifier name

LogFileLocation = Location of the log file:

□ Windows: *SIEBEL_ROOT\log\rpt1_create_view.log*

□ UNIX: *\$SIEBEL_ROOT/log/rpt1_create_view.log*

ScriptLocation = Location of the script

□ Windows: *DBSRVR_ROOT\database_platform\create_view_vod_ver_view.sql*

□ UNIX: *DBSRVR_ROOT/database_platform/create_view_vod_ver_view.sql*

where *database_platform* is the database-type directory name. UNIX example:

/usr/siebel/sea7xx/dbsrvr/DB2UDB/create_view_vod_ver_view.sql

3 Review the *rpt1_create_view.log* and *rpt1_create_view.txt* files for errors.

Running the Report

A script for running the report is included in Release 7.8.2. The report file is created in UTF-8 format.

To run the report

- 1 Navigate to the following directory:

Windows: *SI EBEL_ROOT\bin*

UNIX: *\$SI EBEL_ROOT/bin*

- 2 Type the following command:

- Windows:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_run.txt /h /separator /
/L LogFileLocation\rpt1_run.log ScriptLocation\vod_differ_rpt.sql
```

- UNIX:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_run.txt /h /separator /
/L LogFileLocation\rpt1_run.log ScriptLocation\vod_differ_rpt.sql
```

- z/OS:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_run.txt /q SchemaQualifier /h /separator /
/L LogFileLocation\rpt1_run.log ScriptLocation\vod_differ_rpt.sql
```

where:

Tableowner = Siebel Database tableowner account name

Password = Tableowner password

ODBCDataSource = Data source name for the Siebel Database

OutputFileLocation = Location of the output file:

- Windows: *SI EBEL_ROOT\log\rpt1_run.txt*

- UNIX: *\$SI EBEL_ROOT/log/rpt1_run.txt*

SchemaQualifier = DB2 schema qualifier name

LogFileLocation = Location of the log file:

- Windows: *SI EBEL_ROOT\log\rpt1_run.log*

- UNIX: *\$SI EBEL_ROOT/log/rpt1_run.log*

ScriptLocation = Location of the script:

- Windows: *DBSRVR_ROOT\database_platform\vod_differ_rpt.sql*

- UNIX: *DBSRVR_ROOT/database_platform\vod_differ_rpt.sql*

where *database_platform* is the database-type directory name. UNIX example:
/usr/siebel/sea7xx/dbsrvr/DB2UDB/vod_differ_rpt.sql

- 3 Review the *rpt1_run.log* file for errors.

Reviewing the Report

To review the report, import rpt1_run.txt into Microsoft Excel. The text is ^ (circumflex) delimited.

To review the report

- 1 Start Microsoft Excel and open rpt1_run.txt.
- 2 In the wizard or the dialog boxes that display on how to format the file, make the following selections:
 - Original data type: Delimited.
 - Start import at row: 1.
 - File origin: Accept the displayed code page.
 - Delimiters: Remove the check mark from Tab and place a check mark in Other. In the adjacent box, enter ^ (circumflex).
 If the circumflex is not on your keyboard, press and hold the left Alt key and enter 94 on the numeric keypad. Then release the Alt key.
 - Treat consecutive delimiters as one: Verify this check box does not contain a check mark.
 - Text qualifier: Accept the listed default.
 - Column data format: General.

The imported text file displays in column format. The first row lists the column heads.
- 3 Save the file in .xls format.
- 4 Increase column widths as required to display all the text in each field.

5 Review the report.

The report lists customizations you have made to seed data since installing Release 7.8.1. Use [Table 18](#) to review the report.

Table 18. Columns in the Seed Data Customizations Report

Column	Explanation
Type	The seed data object type. Object types map to UI applet names and include: <ul style="list-style-type: none"> ■ Variable Definition ■ Variable Map Mode ■ Child Variable Map ■ Variable Source ■ Variable Srce Param ■ Signal Action ■ Signal Property ■ Action Parameter
Name	Corresponds to record names in the UI
Action	<ul style="list-style-type: none"> ■ New. You added this object after Release 7.8.1 was installed. ■ Modified. You modified this object after Release 7.8.1 was installed. ■ Deleted. You deleted this object after Release 7.8.1 was installed.
Roottype	These columns list the family tree of the object. The columns provide both the object type and object name. The Roottype and Rootname are the top of the family tree. The lowest level of the tree is greatgrandchildtype and greatgrandchildname. To determine the position in the family tree of an object, do the following: <ol style="list-style-type: none"> 1 Note the object's type in the type column. If this is the same as the roottype, the object is at the top of the tree. 2 If type and roottype are not equal, then check childtype, grandchildtype and so on until you find the type. Use this method to construct the family tree of an object.
Rootname	
childtype	
childname	
grandchildtype	
grandchildname	
greatgrandchildtype	
greatgrandchildname	
Field_ <i>n</i>	
Field_ <i>n</i> _OldVal	The Release 7.8.1 value for this field. If this value blank, it may mean that you have created a new object.

Table 18. Columns in the Seed Data Customizations Report

Column	Explanation
Field_n_NewVal	The current value of this field. If blank, it may mean that you deleted this object. By looking at the Action field and then comparing the OldVal and NewVal fields you can determine how an object was customized.
Y	Ignore this column, if present.

Dropping the Report View

A script for dropping the report database view is included in Release 7.8.2.

To drop the report view

- 1 Navigate to the following directory:

Windows: *SIEBEL_ROOT\bin*

UNIX: *\$SIEBEL_ROOT/bin*

- 2 Type the following command:

- Windows:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_drop_view.txt /h /separator /
/L LogFileLocation\rpt1_drop_view.log ScriptLocation\drop_view_vod_ver_view.sql
```

- UNIX:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_drop_view.txt /h /separator /
/L LogFileLocation\rpt1_drop_view.log ScriptLocation/drop_view_vod_ver_view.sql
```

- z/OS:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /c REM
/O OutputFileLocation\rpt1_drop_view.txt /q SchemaQualifier /h /separator /
/L LogFileLocation\rpt1_drop_view.log ScriptLocation/drop_view_vod_ver_view.sql
```

where:

Tableowner = Siebel Database tableowner account name

Password = Tableowner password

ODBCDataSource = Data source name for the Siebel Database

OutputFileLocation = Location of the output file:

- Windows: *SIEBEL_ROOT\log\rpt1_drop_view.txt*

❑ UNIX: `$SIEBEL_ROOT/Log/rpt1_drop_view.txt`

SchemaQualifier = DB2 schema qualifier name

LogFileLocation = Location of the log file:

❑ Windows: `SIEBEL_ROOT\Log\rpt1_drop_view.log`

❑ UNIX: `$SIEBEL_ROOT/Log/rpt1_drop_view.log`

ScriptLocation = Location of the script:

❑ Windows: `DBSRVR_ROOT\database_platform\drop_view_vod_ver_view.sql`

❑ UNIX: `DBSRVR_ROOT/database_platform/drop_view_vod_ver_view.sql`

where *database_platform* is the database-type directory name. UNIX example:
`/usr/siebel/sea7xx/dbsrvr/DB2UDB/drop_view_vod_ver_view.sql`

Review the `rpt1_drop_view.log` and `rpt1_drop_view.txt` files for errors.

Preparing Mobile User Data for the Database Upgrade

Upgrades: All upgrades.

Environments: All environments.

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 4, "How to Perform the 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.

To prepare mobile users for the database upgrade

- 1 Perform a partial synchronization for mobile users, sending all transactions to the Siebel Database Server.
- 2 Verify that Mobile Web Clients have synchronized and that all changes have been merged into the Siebel Database Server 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 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.
 - c Verify that Workflow Monitor and Workflow Action agents have processed all pending requests. If Workflow Manager has completed successfully, the `S_ESCL_REQ` table should not have any rows.
- 3 To prevent synchronization of Mobile Web Clients with the Siebel Database Server, stop or disable all Siebel Remote components on all Siebel Servers, as described in *Siebel Remote and Replication Manager Administration Guide* and in *Siebel System Administration Guide*.
 - 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 Release 7.5.x 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 IBM 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 Up Campaign Status Values for Siebel Marketing Upgrade

Upgrades: All upgrades.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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 Release 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 19 lists the standard Status values for each object.

Table 19. Standard Status Values

Object	Standard Status Values
Program Plan	Planned Active Completed
Stage Plan	Planned Active Completed
Campaign Plan	Planned Active Completed
Campaign	Planned Active Completed
Campaign Occurrence	Pending Active Finished Manual Error Cancelled

Table 20 lists search specifications for locating records containing Status values. Release 7.5 terminology is used for the object names.

Table 20. 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 TEMPL_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

Preparing Address Data for Upgrade

Upgrades:

- Release 6.2.1 and 7.0.x of Siebel Financial Services applications to Release 7.8.x of Siebel Industry applications.
- Release 7.8.x of Siebel Business Applications to Release 7.8.x of Siebel Industry Applications.

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.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The way address data is stored is changed at Release 7.8. To prepare for the revised storage scheme, you must verify that there are no records with the same row IDs across the tables S_ADDR_PER, S_ADDR_ORG, and S_CON_ADDR.

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\DB2390`
UNIX: `DBSRVR_ROOT/DB2390`
- 2 Review the output generated by the script.
- 3 If the output contains records with duplicate row IDs, revise the `ROW_ID` value of one of the records.
- 4 After addressing all the duplicate row IDs, rerun the script and verify there are no more duplicates.

Preparing Products for Upgrade

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Customizable Products in Work Spaces

The upgrade does not migrate unreleased customizable products in work spaces. If you want to migrate customizable products, you must release them prior to the upgrade. This includes products with components and products with attributes.

Class Products

Verify that the Orderable flag is not set for class products. With this flag unset, 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.

Preserving Marketing Segment Descriptions

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When upgrading from a prior release of Siebel Marketing to Release 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.

Identifying and Resolving Duplicate Row IDs

Upgrades: Release 6.2.1 only.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

During the upgrade from Release 6.2.1 to 7.8, 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 identify and 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\DB2390*

UNIX: *DBSRVR_ROOT/DB2390*

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: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

For upgrades from Release 6.2.1 to 7.8, 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 Siebel Professional Services.

Preparing Forecasting Data for Upgrade

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

For upgrades from Release 6.2.1 to 7.8, 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 Products and Quotes for Upgrade

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Customizable Products

Determine which of your Release 6.2.1 configuration solutions are to be upgraded to Customizable Product quote items in Release 7.8. 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 6.2.1 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 6.2.1 default behavior is to populate this field with \$0.00, and the user must then clear this before entering a header-level discount.

In Release 7.8, 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
```

Verifying Household Data Integrity

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If you choose to implement the new household design, you must run the household verification script (HH_MIG_populate.sql) before upgrading to Release 7.8.

If you choose not to upgrade to the new household design, do not run the script. Also, make sure that the temporary table TEMP_HH_OBJ_MIG does not exist in the database.

The household verification script verifies household data integrity. 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 script ensures that after the upgrade, you will see the same or more household data than 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. The script generates a report based on an output file. Output is in the form of row IDs.

If there is no output, this means Siebel 6.2.1 data integrity is good, and no action is required. If you receive output, this means that no contact is associated with a household for a particular entity.

To run the household verification utility

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_Mi g_popul ate. txt
LogFileLocati on = Location of the log file :
- ❑ Windows: *SIEBEL_ROOT\log\HH_Mi g_popul ate. log*
- ❑ UNIX: \$SIEBEL_ROOT/Log/HH_Mi g_popul ate. log
ScriptLocati on = Location of the script:
- ❑ Windows: *DBSRVR_ROOT\database_pl atform\HH_Mi g_popul ate. sql*
- ❑ UNIX: *DBSRVR_ROOT/database_pl atform/HH_Mi g_popul ate. sql*

Windows example:

```
odbcsql /U Tableowner /P Password /S ODBCDataSource /a /c REM /separator / /O  
C:\sea7xx\si ebsrvr\Log\HH_Mi g_popul ate. txt /L  
C:\sea7xx\si ebsrvr\Log\HH_Mi g_popul ate. log  
C:\sea7xx\dbsrvr\DB2UDB\HH_Mi g_popul ate. sql /v y
```

- 2 If you receive output, review the temporary table and verify 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.

At least one contact associated with an entity must also be associated with the household for that entity.

Preparing Financial Services Application Tables for Upgrade

Upgrades: Release 6.2.1 only.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

DESC_TEXT is a column in two tables, S_CONDI TION and S_PROD_BNFT. To broaden platform support, this column is reduced in length from 250 to 200 characters during the upgrade from Release 6.2.1 to Release 7.8.

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.

Preparing Siebel Configurator Data for Upgrade

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 upgraded properly, customers must have appropriate model product associations. For more information on this and other recommended preupgrade tasks, see *Siebel Interactive Selling Applications Upgrade Guide*.

Setting the Value of S_SRC_PAYMENT.TYPE_CD

Upgrades: Release 7.0.x.

All environments. This topic applies to all Siebel environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 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.

9

Preparing a Development Environment for Upgrade

This chapter describes the steps in preparing a development environment for upgrade. It includes the following topics:

- [“Prerequisites for a Development Upgrade”](#)
- [“About Moving Tables”](#)
- [“Checking In Development Repository Projects”](#) on page 152
- [“Preparing EIM Tables for Upgrade”](#) on page 153
- [“Preparing for Intersection Table Maintenance and Deduplication”](#) on page 155
- [“Determining Which Template File Was Used During an Extract or Merge”](#) on page 156
- [“Renaming the Siebel Repository Using Siebel Tools”](#) on page 157

Prerequisites for a Development Upgrade

Upgrades: All upgrades.

Environments: Development environment only.

Before you upgrade your development environment, make sure that the development database configuration meets the database requirements outlined in [“Verifying Database Server Configuration”](#) on page 117, and meets the requirements depicted in the *Siebel Installation Guide* for the operating system you are using.

If your development environment platform is DB2 UDB for Windows and UNIX, see *Upgrade Guide*.

NOTE: Make sure that the development database uses binary sort order. Production databases are not constrained by this requirement.

If you have not already done so, copy the Upgrade Planning Worksheet, located in [Appendix C, “Upgrade Planning Worksheet,”](#) and fill out the appropriate fields with the information you will need to perform the upgrade. Contact your database administrator or systems programmer for help in completing the worksheet. Also, refer to [“Preparing to Run the Database Server Configuration Utility”](#) on page 164 for a description of the information you are required to enter when you run the Database Server Configuration Utility to perform upgrade operations.

About Moving Tables

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Some of the tables are going to have columns added to them as part of the upgrade. When this happens, the length of the record will increase causing it to require a larger tablespace or bufferpool. Moving the identified tables to a new, larger and differently named tablespace allows you to maintain the paradigm of multiple tables per tablespace which is present for all pre-7.7 schemas.

Before you move the tables, you need to drop any standard or custom views. Complete the following procedure to find the views that are defined on a table.

NOTE: The example used in the procedure assumes that you are creating a new 16K tablespace within the same database as the older, smaller tablespace.

To find views that are defined on a table

- Run the following SQL statement to produce the list of views:

```
select * from sysibm.sysVIEWdep
WHERE Bcreator = 'yourschema'
AND BNAME IN ('S_ASSET', 'S_PROD_INT');
```

This SQL statement produces the drop statements into a sequential dataset:

```
SELECT DISTINCT 'DROP VIEW ' || '*** yourschema.' || V.DNAME || ' ';
FROM SYSIBM.SYSVIEWDEP V
WHERE V.BNAME IN ('S_ASSET', 'S_PROD_INT')
WHERE V.BNAME = 'S_ASSET'
AND BCREATOR = 'yourschema';
```

In order to process these drop statements on the z/OS host, you need to create a PDS member with the output from the preceding statement. Then submit this member through DSNTEP2 using JCL.

You can also list these tables in the override.inp file. This will cause the tables to be recreated in a new database with one tablespace and one table.

If you are using the preconfigured storage control file, you do not need to move any tables as a preupgrade task. This is because the existing schema is not being preserved and all tables will be recreated in the 1:1:1 model.

Checking In Development Repository Projects

Upgrades: All upgrades.

Environments: Development environment only.

Developers who are using Siebel Tools with a local SQL Anywhere database must check in their projects to the development repository.

Make sure that all project locks in your current Siebel repository have been released to prevent inadvertent loss of development work during the upgrade.

If you are using Siebel Workflow Manager, you must run the Workflow Monitor Agent and Workflow Action Agent to completion before upgrading to Release 7.8. The S_ESCL_REQ table should not have any rows if Workflow Manager has completed successfully.

Preparing EIM Tables for Upgrade

Upgrades: All upgrades.

Environments: Development environment only.

This section describes how to drop EIM tables and prepare S_ASSET_POSTN and S_ETL_TIME_DAY for upgrade.

To drop EIM tables and prepare S_ASSET_POSTN and S_ETL_TIME_DAY for upgrade

- 1 Navigate to the `DBSRVR_ROOT\db2390\` (Windows) or `DBSRVR_ROOT/db2390/(UNIX)` directory and open the `predevftp.txt` file.
- 2 Edit the `predevftp.txt` file using the following information:
 - a Change `&IP` to the IP address of your DB2 host, for example, `ZM01`.
 - b Change `&Username` to the your own user name, for example, `sadm n`.
 - c Change the constant `SIEBELQ1` to your own dataset high-level qualifier (*DSNHLQ*), for example, `sadm n`, using a command like `Replace All`.

Use the same high-level qualifier for all z/OS upgrade datasets. The *DSNHLQ* must be 18 characters or less and can have multiple nodes separated by periods. A node must be one to eight characters in length. The *DSNHLQ* may be the TSO ID.
 - d Change `&SiebelRoot` to your SIEBEL installation path, for example, `D:\7.8\`.
- 3 After modifying the `predevftp.txt` file, save it, and then close the file.
- 4 Double-click `ftpdev.bat` (located in `DBSRVR_ROOT\db2390`) or if you are using a UNIX operating system, issue the following command: `Ftp -i > ftpfile.bat`.

You are prompted to enter the password for the user name that you entered in `predevftp.txt` to replace `&Username`.
- 5 Enter the password, and then press `Enter`.

All files generated to this point of the upgrade process transfer from the midtier to the z/OS host.
- 6 After the transfer is successful, log on to the mainframe, and navigate to the *DSNHLQ*. `PREDEVO.JCL` file, for example, `SIEBELQ1.PREDEVO.JCL`.
- 7 In the job card set the following parameters:

```
CLASS=S
MSGCLASS=X
```

```
JOBPARAM S=YOUR_HOST_NAME
```

If your Host server is QA01, the job card should look like the following:

```
///###PREDV JOB ACCNT#, CLASS=S, MSGCLASS=X, MSGLEVEL=(1, 1),
//          TIME=1440, REGION=OM, NOTIFY=&SYSUID, LINES=9999
//*
/*JOBPARM S=QA01
```

8 From the command line, execute the following commands:

- C SIEBELQ1 *DSNHLQ* all

Where:

DSNHLQ is the high level qualifier you specified in [Step 2 on page 153](#).

- C <SCHEMA> *YOUR_DB2_SCHEMA*

For example,

```
C <SCHEMA> DB2QUAL ALL
```

- C <DEVHOST> *YOUR_HOST_LPAR*

For example,

```
C DEVHOST QA01 ALL
```

- C <DB2LIB1> *YOUR_DB2_LOAD_DSN*

For example,

```
C DB2LIB1 DSN710.SDSNLOAD ALL
```

- C <DB2LIB2> *YOUR_SECOND_DB2_LOAD_DSN*

For example,

```
C <DB2LIB2> DSN710.RUNLIB.LOAD ALL
```

- C <DB2SYS> *YOUR_DB2_SUBSYSTEM*

For example,

```
C <DB2SYS> Q10K ALL
```

9 After submitting the job, enter cancel on the command line or press PF3 to save changes.

10 Verify that the job ran successfully.

- a Verify that the RC=0.

- b Verify that the following datasets were created.

NOTE: If you are upgrading from a FINS 7.0.4 application, you need to use the SIS 7.0.4 library.

- For upgrades of Siebel Business applications:

DSNHLO. PRE. HOR621. JCLLI B

DSNHLO. PRE. HOR704. JCLLI B

DSNHLO. PRE. HOR752. JCLLI B

- For upgrades of Siebel Industry applications:

DSNHLO. PRE. SI A621. JCLLI B

DSNHLO. PRE. SI A752. JCLLI B

DSNHLO. PRE. SI S63. JCLLI B

DSNHLO. PRE. SI S704. JCLLI B

- 11 Edit the dataset from the list above that applies to your upgrade path.

NOTE: Each library has an @@README member to guide you through what to submit.

- 12 Submit the jobs as instructed in the @@README file.

- 13 Verify that the job ran successfully, RC=0 or RC=4.

- 14 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Preparing for Intersection Table Maintenance and Deduplication

Upgrades: All upgrades.

Environments: Development environment only.

Perform the following task to create the storage_dedup.ctl file. This file is used for deduplication procedures and intersection table maintenance.

To create a storage_dedup.ctl file

- 1 Create a copy of the storage_dedup.ctl file located in the *DBSRVR_ROOT\DB2390* directory.

- 2 Rename the copy of storage_dedup.ctl to preserve it as a backup, for example, save it as *ORIGstorage_dedup.ctl*.

- 3 Navigate to the *DBSRVR_ROOT\DB2390* directory and open *dbconf.xls*.

NOTE: If your midtier is UNIX, then the *dbconf.xls* and *storage_dedup.ctl* files need to be transferred to Windows using the *ftp* command. The *dbconf.xls* file needs to be transferred in *binary mode* while the *storage_dedup.ctl* file needs to be in *ASCII mode*. Using these transfer modes prevents file corruption.

- 4 Select the HOME tab page, and click Import.

- 5 When you are prompted to select the storage_dedup file, navigate to the *DBSRVR_ROOT\DB2390* directory and select *storage_dedup.ctl*.

The default values screen appears.

- 6 On the default values screen, enter the values for your system:
 - a Enter your values for the following parameters:
 - ❑ Table Storage Group for Table
 - ❑ Index Storage Group for Indexes
 - ❑ 4-KB Buffer Pool Name
 - ❑ 16-KB Buffer Pool Name
 - ❑ Index Buffer Pool Name
 - b For the Database Name Prefix, replace the placeholder, *SIDB*, with your database name prefix. Your database name prefix must be a maximum of four characters, for example *nnnn*, where *nnnn*. This may be the last four letters of your tableowner ID.
 - c For the Encoding Scheme, specify your database encoding scheme, either ASCII or EBCDIC.After entering and verifying all values, click Set.
- 7 Return to the HOME tab page, and click Export.

Save this database configuration as `storage_dedup.ctl` in the `DBSRVR_ROOT\DB2390` directory.

Determining Which Template File Was Used During an Extract or Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This section describes how to determine which template file that was used for an extract/merge when a storage control file is created.

To determine the template file that was used for an extract/merge

- 1 Open the `dbextract.log` file and review the command line that was entered.
- 2 Check the value for the `/i` parameter. You can also check the `upgwi z.log` file for the `strgxrct.exe` command string.

```
/i D:\18025\dbsrvr\db2390\storage_p_e.ctl
```

```
2004-01-28 09:10:32D:\18025\si_ebsrvr\bin\strgxrct.exe /u QADMIN /p ***** /c Q10K /d yourschema /I D:\18025\si_ebsrvr\log\dbconfi_g_extract_merge_mf_lk5\output/dbextract.log /i D:\18025\dbsrvr\db2390\storage_p_e.ctl /o D:\18025\dbsrvr\db2390/SStempstore.txt /v D:\18025\dbsrvr\db2390\overri de.inp /8 D029 /9 SIEBTS /1 SYSDEFLT /2 SYSDEFLT /3 BP1 /4 BP2 /7 EBCDIC
```

Renaming the Siebel Repository Using Siebel Tools

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Your upgrade will encounter errors if you have more than one existing repository for a development environment upgrade. Export and archive any redundant repositories from your environment before you upgrade your development environment.

Four separate repositories are used during the development upgrade process:

- Your existing development repository
 - 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*.
- Three new repositories which are automatically loaded when you run the Siebel Upgrade Wizard.
 - Prior standard repository
 - New standard repository (Release 7.8)
 - New customer repository (Release 7.8) which, after the merge, becomes the customized 7.8 repository.

Before these repositories are loaded, you must verify that your existing repositories do not use the names reserved for the upgrade process (see [Step 7](#) below).

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.** Your new custom repository, which results from the merge, and includes your custom configurations.
 - **New Siebel Repository.** The new Siebel standard repository.
 - **Prior vX.x Siebel Repository.** The prior Siebel Repository.

8 Locate your current Siebel Repository in the list applet.

9 Click on the name and change it to Prior Customer 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.

For more information about renaming repositories, see *Configuring Siebel Business Applications*.

TIP: Prefix the repository name with a date; for example, 20031202 Prior v6.2.1 Customer Repository.

10 Step off the list to commit the record to the database.

If no repository is named Prior Customer Repository, the upgrade process renames the Siebel Repository to Prior Customer Repository.

10 Preparing a Production Environment for Upgrade

This chapter describes the steps in preparing a production environment for upgrade. It includes the following topics:

- [“Prerequisites for a Production Upgrade”](#)
- [“Transferring the Customized Repository and Schema Definition Files” on page 161](#)
- [“Renaming the Production Environment Repository” on page 162](#)

Prerequisites for a Production Upgrade

Upgrades: All upgrades.

Environments: Production test, production.

NOTE: You must be thoroughly familiar with the upgrade process before beginning the production upgrade. Before upgrading your production environment, perform a test upgrade in your production test environment to familiarize yourself with the process and to eliminate errors that can affect upgrade success or performance.

Before beginning the upgrade of your production environment, verify that the production database configuration meets the database requirements outlined in the section [“Verifying Database Server Configuration” on page 117](#).

NOTE: Verify that your production database uses binary sort order.

About Subsystems and Tableowners

[Table 21](#) describes the subsystem and tableowner requirements for the production upgrade.

Table 21. Production Upgrade Subsystem and Tableowner Requirements

Subsystem	Tableowner
One subsystem	Two tableowners
Two subsystems	One tableowner
Two subsystems	Two tableowners

If you have not already done so, copy the Upgrade Planning Worksheet, located in [Appendix C, “Upgrade Planning Worksheet,”](#) and fill out the appropriate fields with the following information. Contact your database administrator or systems programmer for help in completing the worksheet. Also, refer to [“Preparing to Run the Database Server Configuration Utility” on page 164](#) for a description of the information you are required to enter when you run the Database Server Configuration Utility to perform upgrade operations.

Source Database Values

For the production environment upgrade, you need the following values for your source database:

- **ODBC Data Source Name.** This name must be the same as the name of the database qualifier.

NOTE: On Microsoft Windows, the Siebel Server installation process automatically creates the data source, using the format `Si ebSrvr_EnterpriseName`. To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > ODBC data source. Click the System DSN tab and you will find the name of your ODBC data source.

- **Database User Name.** The ID allowed to log into the Siebel Database. This user ID should have authorization to set CURRENT SQLID and must be set up as an employee within your Siebel Business application. The employee record must have the Siebel administrator responsibility. SADMIN is the default administrator user name and password, but you can use any name that has SADMIN credentials. If this user does not already exist in your database, or does not have Siebel administrator privileges, then you must add this to your database prior to proceeding with the upgrade. For more information, see [“Adding a Siebel User” on page 132](#).

This login must be the same as the target database user name that you will specify in the upgrade configuration utility.

- **Database Password.** The password for the ID used to log into the Siebel Database.
- **Schema Qualifier.** The up to eight-character identifier that designates Siebel Schema. This is also an authorization ID. The schema qualifier must start with a letter, cannot contain special characters, and must be entered in uppercase.
- **Group Authorization ID.** The user ID of the group to whom schema access is granted, for example, SSEROLE.

Target Database Values

For the production environment upgrade, you need the following values for your target database:

- **Schema Qualifier.** The up to eight-character identifier that designates Siebel Schema. This is also an authorization ID. The schema qualifier must start with a letter, cannot contain special characters, and must be entered in uppercase.
- **DB2 Subsystem.** Verify that there are no objects in the target DB2 subsystem before you start the upgrade.

- **Database User Name.** The ID allowed to log into the Siebel Database, for example, SBLDBA. This user ID should have authorization to set CURRENT SQLI D and must be set up as an employee within your Siebel Business application. The employee record must have the Siebel administrator responsibility. SADMIN is the default administrator user name and password, but you can use any name that has SADMIN credentials. If this user does not already exist in your database, or does not have Siebel administrator privileges, then you must add this to your database prior to proceeding with the upgrade. For more information, see [“Adding a Siebel User” on page 132](#).

This login must be the same as the target database user name that you will specify in the upgrade configuration utility.

- **Database Password.** The password for the ID used to log into the Siebel Database.
- **Group Authorization ID.** The user ID of the group to whom schema access is granted, for example, SSEROLE.
- **Storage Group for Indexes.** The name of the storage group used for temporary indexes. The default value is SYSDEFAULT.
- **Storage Group for Tables.** The name of the storage group used for temporary tables. The default value is SYSDEFAULT.
- **ODBC Data Source Name.** This name must be the same as the name of the database qualifier.

The remaining information required for upgrade will be derived from your prior source database layout.

Transferring the Customized Repository and Schema Definition Files

Upgrades: All upgrades.

Environments: Production test and Production environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When you upgraded your development environment, the new customized repository was exported to a file called custrep.dat and the modified schema definition was exported to a file called schema.ddl in the *DBSRVR_ROOT\DB2390* directory on the Siebel Server client machine from which you ran the upgrade.

Before doing a production or production test environment upgrade, you must copy the upgraded repository definition files (custrep.dat and schema.ddl) from the development environment to the production environment.

NOTE: If you modified repository objects or schema definitions after completing the development upgrade (upgphys), you must regenerate the schema.ddl and custrep.dat files. See [“Regenerating the Repository Definition Files” on page 295](#).

To move the repository and schema definition files to the production environment

- 1 In the development environment, navigate to the following directory:

Windows: `DBSRVR_ROOT\DB2390`

UNIX: `DBSRVR_ROOT\DB2390`

- 2 Copy the following files:

custrep.dat

schema.ddl

- 3 In the production environment, put these files in the following location:

Windows: `DBSRVR_ROOT\DB2390`

UNIX: `DBSRVR_ROOT\DB2390`

The custrep.dat file will be used by the Siebel Upgrade Wizard to import the New Customer Repository. The schema.ddl file will be used by the Siebel Upgrade Wizard to create the new database schema.

Renaming the Production Environment Repository

Upgrades: All upgrades.

Environments: Production test, production.

Two separate repositories are used during the production upgrade process:

- **Your existing production repository.** To prevent a naming conflict, before you run the upgrade, rename your existing production repository (*Siebel Repository*) to *Prior Customer Repository*. After the upgrade, your new Release 7.8 production repository is given the name *Siebel Repository*.

Rename your existing production repository following the procedure described in [“Renaming the Siebel Repository Using Siebel Tools”](#) on page 157.

- **New Customer Repository.** This repository is loaded when you run the Siebel Upgrade Wizard.

CAUTION: Your upgrade will encounter errors if you have more than one existing repository for a production upgrade. Export and archive any redundant repositories from your environment before you upgrade your production environment.

For further information about renaming repositories, see *Configuring Siebel Business Applications*.

11 Upgrading the Database

This chapter describes the steps that are common to all database upgrades, for example, using the Database Server Configuration utility to perform the upprep and upgphys portions of an environment upgrade. Refer to the roadmap for your upgrade in [Chapter 4, “How to Perform the Upgrade”](#) for a complete list of all the tasks you must perform to upgrade to Siebel 7.8.

This chapter contains the following topics:

- [“Changing the Database Server Configuration Utility Language”](#)
- [“Preparing to Run the Database Server Configuration Utility” on page 164](#)
- [“Running the Database Server Configuration Utility Under Windows” on page 168](#)
- [“Running the Database Server Configuration Utility Under UNIX” on page 169](#)
- [“Starting the Siebel Upgrade Wizard” on page 170](#)
- [“Stopping the Siebel Upgrade Wizard” on page 172](#)
- [“Migrating Address Data from Custom Extension Columns” on page 173](#)
- [“Preparing for a Non-Development-Environment Upgrade” on page 180](#)
- [“Installing New License Keys After Upgrade” on page 180](#)
- [“Deleting Redundant Upgrade Files” on page 181](#)

Changing the Database Server Configuration Utility Language

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The Database Server Configuration utility launches in the language selected when you ran the Siebel Enterprise System 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 utility 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 utility, you first need to install the appropriate language pack on the Siebel 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.

For more information about the Database Server Configuration utility, see [“About the Database Server Configuration Utility” on page 37](#).

Preparing to Run the Database Server Configuration Utility

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Use this topic to identify the information you must enter when running the Database Server Configuration utility. Collect this information and verify it before running the utility.

The Database Server Configuration utility requests information about the upgrade process you want to perform. It then adds this information to a master upgrade file and calls an SQL generator. The SQL generator uses the information to create or populate SQL files.

The Siebel Upgrade Wizard runs after the Database Server Configuration utility exits. The wizard executes the SQL files against the Siebel Database.

Under Windows, the Database Server Configuration utility is a wizard that displays a series of windows. You enter the requested information in each window. Most of the windows contain a Help button. When you click the Help button, an explanation of what to enter in the wizard appears.

Under UNIX, the Database Server Configuration utility displays as a set of screens in a shell command window. You enter the requested information at the prompt.

[Table 22](#) lists the information that you must enter in the Database Server Configuration utility.

[Table 23 on page 167](#) lists additional information the utility requires when you perform a production environment upgrade.

For more information on the Database Server Configuration utility, see [“About the Database Server Configuration Utility” on page 37](#).

Table 22. Information Required for the Database Server Configuration Utility

Screen Name	Required Information
Siebel Gateway Name Server Address	Windows only. The Siebel Gateway Name Server machine name and the Enterprise Server name.
Siebel Server Directory	The absolute path of the directory where the Siebel Server is installed, for example, D:\sea78\si ebsrvr. For UNIX, do <i>not</i> enter the string \$SIEBEL_ROOT.
Siebel Database Server Directory	The absolute path of the directory where the Siebel Database Server is installed—for example D:\sea_7x\dbsrvr.
RDBMS Platform	Choose IBM DB2 UDB for z/OS.

Table 22. Information Required for the Database Server Configuration Utility

Screen Name	Required Information
Siebel Database Operation	Choose Upgrade Database. The other menu choices are for database installation and administration.
Environment Type	<ul style="list-style-type: none"> ■ Choose Development for development environment upgrades. ■ Choose Production for production and production test environment upgrades.
Upgrade Options	Choose one of the following: <ul style="list-style-type: none"> ■ Upgrade Siebel Database Schema (upgrep): development upgrades only ■ Upgrade Siebel Database Schema (upgphys): development upgrades only ■ Prepare for Production Upgrade: production upgrades only ■ Upgrade Siebel Database Schema (upgrep + upgphys): production upgrades only
Siebel Industry Application	<p>This screen appears if you are upgrading from a Siebel Industry Solution or Siebel Industry application.</p> <p>Choose the application you are upgrading from.</p> <p>If you have upgraded to the base Siebel Business application as part of upgrading to the new Siebel Industry application release, choose Siebel Business application (HOR).</p>
Current Siebel Version	Choose the application version you are upgrading from.
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.
Source Database ODBC Data Source Name	<p>Verify the ODBC name for connecting to the source Siebel Database for your upgrade. If it is not correct, enter the correct ODBC name.</p> <p>The ODBC data source <i>must</i> have the same name as the subsystem. When you set up the ODBC connection in DB2 Connect, use the actual subsystem name for the database alias.</p> <p>Windows: To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DNS tab to find the name of your ODBC data source.</p> <p>UNIX: To find the name of your ODBC data source, type: vi \$ORACLE_HOME/dbs/odbc.ini</p>

Table 22. Information Required for the Database Server Configuration Utility

Screen Name	Required Information
Source Database User Name	<p>Enter the source user name and password for the Siebel administrator of the source database for your upgrade.</p> <p>NOTE: The source database user name (user ID) needs to have authorization to set CURRENT SQLID.</p>
Source Siebel Schema Qualifier	<p>Enter the eight-character identifier that designates the Siebel Schema for your source database. This is also an authorization ID. The schema qualifier must start with a letter, cannot contain special characters, and must be entered in uppercase.</p>
Target Database ODBC Data Source Name	<p>Verify the ODBC name for connecting to the target Siebel Database for your upgrade. If it is not correct, enter the correct ODBC name.</p> <p>The ODBC data source <i>must</i> have the same name as the subsystem. When you set up the ODBC connection in DB2 Connect, use the actual subsystem name for the database alias.</p>
Target Database User Name	<p>Enter the target user name and password for the Siebel administrator of the target database for your upgrade.</p> <p>NOTE: The target database user name (user ID) needs to have authorization to set CURRENT SQLID.</p>
Target Siebel Schema Qualifier Target Security Group ID	<p>Enter the following parameter values:</p> <p>Target Schema Qualifier. Enter the up to eight-character identifier that designates Siebel Schema for your target database. This is also an authorization ID. The schema qualifier must start with a letter, cannot contain special characters, and must be entered in uppercase.</p> <p>Target Security Group ID. Enter the user ID of the group to whom schema access is granted, for example, SSEROLE.</p>
Storage Group for Temporary Indexes Storage Control File	<p>Enter the following parameter values:</p> <p>Storage Group for Temporary Indexes. Enter the name of the storage group provided by the database administrator (the default value is SYSDEFLT).</p> <p>Storage Control File. Use either of the following methods to select the storage control file that contains the previously customized database storage layout (this is the file that you prepared in “Preparing the Storage Layout of the Schema” on page 119):</p> <ul style="list-style-type: none"> ■ Windows: Accept the default value displayed in the Storage Control File field (this is the <code>DBSRVR_ROOT\DB2390</code> directory. For example, <code>D:\DBSRVR_ROOT\DB2390\storage.ctl</code>) or use the Browse button to navigate to a different directory. ■ UNIX: Enter the path to your storage control file.

Table 22. Information Required for the Database Server Configuration Utility

Screen Name	Required Information
Primary Quantity for Temporary Index Space	Enter the primary and secondary quantities for temporary index space.
DDL Commit Frequency	Enter the DDL commit frequency for your upgrade.
DDL Output Directory	Accept the default or enter the directory name. NOTE: When this process is complete, this directory contains all of the files necessary to create the Siebel Database schema. These files must be manually applied by the database administrator.
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, slashes, or symbols in the name of the log output. CAUTION: If you are restarting or recovering a previous incomplete upgrade, do not change the Log Output Directory that you previously selected.

Additional Information Required for Production Upgrades

When you perform a production environment upgrade, the additional information shown in [Table 23](#) is required when you run the Database Server Configuration utility in Prepare for Production Upgrade mode.

Note that several screens request information about the Siebel Database in the development environment, not the production environment.

Table 23. Additional Information Required for Production Upgrades

Screen Name	Required Information
ODBC Data Source Name for Development Database	Windows only. The ODBC name for connecting to the development environment Siebel Database. If you are upgrading without a development environment, this is the ODBC 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.
Import Repository Name	Enter the name of the upgraded repository in Siebel Tools repository in the development environment database. Typically, this is <i>Siebel Repository</i> .

Running the Database Server Configuration Utility Under Windows

Upgrades: All upgrades.

Environments: All environments.

Platforms: Windows only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Run the Database Server Configuration utility to upgrade the Siebel Database. The utility collects information, populates a master configuration file, and calls the SQL generator to create SQL commands. The Siebel Upgrade Wizard then uses the configuration file and SQL commands to generate upgrade files that you apply on the mainframe, and to make upgrade changes to the Siebel Database.

For more information on the Database Server Configuration utility, see [“About the Database Server Configuration Utility” on page 37.](#)

Prerequisite: Collect the information that the Database Server Configuration utility requires. See [“Preparing to Run the Database Server Configuration Utility” on page 164.](#)

To run the Database Server Configuration utility 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 Enterprise Servers 7.x > Configure DB Server.
The first window of the Database Server Configuration utility appears.
- 3 Enter the information requested in each window and click Next.
To return to previous windows, click Previous.
- 4 When the runnow window displays, click OK to apply the configuration. The utility displays a screen that lists the values you have entered.
- 5 To amend any of the configuration values, click Previous to return to the appropriate screen and make changes. Otherwise, click Finish.
- 6 When prompted to start the Siebel Upgrade Wizard, click OK to begin the upgrade. The Database Server Configuration utility calls the SQL generator to create or populate SQL scripts.
To start the upgrade at a later time, click Cancel. See [“Starting the Siebel Upgrade Wizard” on page 170](#) for information on manually restarting the Siebel Upgrade Wizard.

Running the Database Server Configuration Utility Under UNIX

Upgrades: All upgrades.

Environments: All environments.

Platforms: UNIX only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Run the Database Server Configuration utility to upgrade the Siebel Database. The utility collects information, populates a master configuration file, and calls the SQL generator to create SQL commands. The Siebel Upgrade Wizard then uses the configuration file and SQL commands to generate upgrade files that you apply on the mainframe, and to make upgrade changes to the Siebel Database.

For more information on the Database Server Configuration utility, see [“About the Database Server Configuration Utility” on page 37.](#)

Prerequisite: Collect the information that the Database Server Configuration utility requires. See [“Preparing to Run the Database Server Configuration Utility” on page 164.](#)

To run the Database Server Configuration utility 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.

The path must end in siebsrvr. For example, /usr/siebel/sea7xx/siebsrvr

3 Source environment variables:

Korn (do not use a Bourne shell)

. siebenv.sh

C shell

source siebenv.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 Server Configuration utility runs. The value of this variable is a language identifier string. For example, enu is the identifier string for English.

If either the SIEBEL_ROOT or LANGUAGE value is not set or is incorrect, you must correct them before proceeding.

5 Start the Database Server Configuration utility script in either a Korn shell or C shell:

```
./bin/dbsrvr_config.ksh
```

- 6 Enter the information requested in each screen. Click ENTER to proceed to the next screen.
After you have entered all the requested information, the utility displays a screen that lists the values you entered. If you need to make changes, exit and rerun the utility.
- 7 When prompted to start the Siebel Upgrade Wizard, answer Yes to begin the upgrade. The Database Server Configuration utility calls the SQL generator to create or populate SQL scripts.
To start the upgrade at a later time, answer No. See [“Starting the Siebel Upgrade Wizard” on page 170](#) for information on manually restarting the Siebel Upgrade Wizard.

Starting the Siebel Upgrade Wizard

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The Siebel Upgrade Wizard calls drivers that execute SQL scripts against the Siebel database. These scripts were created or populated when you ran the Database Server Configuration utility.

When the Siebel Upgrade Wizard encounters errors that prevent continuing the upgrade, it writes the errors to the log files and stops the upgrade. After you correct the problems causing the errors, you can restart the Siebel Upgrade Wizard. In most cases, the wizard resumes from where it stopped.

For more information on the Siebel Upgrade Wizard, see [“About the Siebel Upgrade Wizard and Driver Files” on page 40](#).

Prerequisites for Restarting the Siebel Upgrade Wizard

If the Siebel 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.
- 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: *SIEBEL_ROOT\log\PROCESS\output*

UNIX: *\$SIEBEL_ROOT/log/PROCESS/output*

Also delete the state.l og file. It is located in the following directory:

Windows: *SIEBEL_ROOT\log\PROCESS\state*

UNIX: *\$SIEBEL_ROOT/log/PROCESS/state*

To manually start the Siebel Upgrade Wizard

- 1 Navigate to the following directory:

Windows: *SIEBEL_ROOT\bin*

UNIX: *\$SIEBEL_ROOT/bin*

- 2 Enter the following command:

Windows: *si ebupg /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 mode, upgrade environment, and the Siebel release from which you are upgrading. The file is located in *SIEBEL_ROOT\bin* (UNIX: *\$SIEBEL_ROOT/bin*).

For example, to start a development upgrade from Release 7.0.4 under Windows, enter the following command:

si ebupg /m master_upgrep_dev_704.ucf.

- 3 To begin the upgrade, click OK (Windows) or click Enter (UNIX).

A check mark appears beside each item as it is completed.

- 4 The Siebel Upgrade Wizard stops after the step, "Export schema definition," and the following message appears:

Pause #1

Please make sure your database administrator performs operations based on the files in the DDL Output Directory to complete this step. Once the operations are completed, please select Yes. To stop now and continue upgrade later, please select No. (To resume, please start the Upgrade Wizard from the command line using option /m master_upgrep_<env>_<path>_mf.ucf.)

- 5 Do not click either Yes or No when the preceding message appears. Depending on whether you are upgrading from a midtier environment or a mainframe environment, continue your upgrade by completing the appropriate tasks:
 - If you are upgrading from a midtier environment, see [“Executing SQL Files After the First Pause” on page 183](#).
 - If you are upgrading from a mainframe environment, see [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 213](#).

Stopping the Siebel Upgrade Wizard

Upgrades: All upgrades.

Environments: All environments.

Platforms: All platforms.

The Upgrade Wizard sends SQL commands to the RDBMS. These commands make schema changes and migrate data. If you stop the Upgrade Wizard and restart it, the Upgrade Wizard resumes the upgrade after the last upgrade step it successfully completed.

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 hung. Some SQL commands issued by the Upgrade Wizard or by its utilities can take considerable time to complete.

If you are not sure whether or not the Upgrade Wizard is hung, contact Siebel Technical Support.

Stopping the Upgrade Wizard can have varying effects on the RDBMS. Before restarting the Upgrade Wizard, review the RDBMS logs. Run SQL commands as needed to resolve errors found in the RDBMS logs.

Stopping the Upgrade Wizard Under Windows

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 completed. There might be a delay while the step completes in the RDBMS.

Stopping the Upgrade Wizard Under UNIX

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 already-issued SQL commands.

Related Topic

[“Starting the Siebel Upgrade Wizard” on page 170.](#)

Migrating Address Data from Custom Extension Columns

Upgrades:

- From Release 6.2.1 of Siebel Financial Services applications on IBM DB2 z/OS systems to Release 7.8.x of Siebel Industry applications on IBM DB2 z/OS platforms
- From Release 7.0.x of Siebel Financial Services applications to Release 7.8 of Siebel Industry applications
- From Release 7.8 of Siebel Business applications to Release 7.8 of Siebel Industry applications

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: Development, production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 Siebel Expert Services if you need assistance.

Prerequisite. You must have run the Database Server Configuration utility to pause # 1. This creates the files required for this procedure.

To migrate Address data from custom extension columns

- 1 Open the ddl.ctf file with a text editor.

NOTE: When you edit any of the upgrade files, they must be saved in UTF-8 encoding. Opening a file using Microsoft Notepad and clicking File > Save automatically saves the existing encoding.

Here is the path to the file in the Siebel Database Server installation:

Windows: `DBSRVR_ROOT\DB2390`

UNIX: `DBSRVR_ROOT/DB2390`

- 2 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 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 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\DB2390\DBOUTPUT\UPGRADE*

UNIX: *DBSRVR_ROOT/DB2390/DBOUTPUT/UPGRADE/*

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'
el se 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

```

To view the data in your custom extension columns, you must configure your application to expose the custom extension columns added to S_CON_ADDR.

- 5 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.ctl` file.

Here is the path to the file in the Siebel Database Server installation:

Windows: `DBSRVR_ROOT\DB2390\upgrade\version`

UNIX: `DBSRVR_ROOT/DB2390/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

```

To view the data in your custom extension columns, you must configure your application to expose the custom extension columns added to S_ADDR_PER.

- 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.

Preparing for a Non-Development-Environment Upgrade

Upgrades: All upgrades.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 non-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\DB2390` directory.
- 4 In the `DBSRVR_ROOT\DB2390` 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 tablespace, separate from the Siebel Database. Install the Siebel Database from the new release in the new tablespace. 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 License Keys After Upgrade

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 development environment upgrades, perform this task after upgrading the Siebel Database schema (`upgrep`).

- For production environment upgrades, perform this task after upgrading the custom database schema (Uppgphys).

To add new license keys

- 1 Start Siebel Tools (current release version) from a production workstation and log on to the Siebel Database Server as the Siebel administrator.
- 2 Add your new license keys.

Deleting Redundant Upgrade Files

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After you complete the database upgrade process, you can delete any upgrade files that are no longer required (for example, source tables). A number of stored procedures are installed during the database upgrade on the z/OS host to facilitate upgrade processing. These stored procedures are not required after the upgrade has been completed successfully and can also be deleted. The procedure in this topic describes how to drop the upgrade stored procedures.

For information on installing the stored procedures, see [“Installing Stored Procedures on the z/OS Host” on page 219.](#)

To drop the upgrade stored procedures

- 1 Log on to the z/OS host.
- 2 Submit the JCL in the dataset member `DSNHLO.SIEBEL.SP.CNTL(DRPSIA)` or `DSNHLO.SIEBEL.SP.CNTL(DRPHOR)`, depending on your upgrade path.
- 3 Press PF3 to complete the process.

12 Upgrading a Development Environment Database from the Midtier

This chapter describes the tasks that are specific to performing a midtier-centric development environment upgrade. Refer to the roadmap for your upgrade in [Chapter 4, “How to Perform the Upgrade”](#) for a complete list of all the tasks you must perform to upgrade your development environment to Siebel 7.8.

This chapter contains the following topics:

- [“Executing SQL Files After the First Pause”](#)
- [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 185](#)
- [“Executing SQL Output Files After the Second Pause” on page 186](#)
- [“Executing SQL Output Files After the Second Pause - Stage 1 of 2” on page 187](#)
- [“Deploying Stored Procedures and User-Defined Functions” on page 188](#)
- [“Executing SQL Output Files After the Second Pause - Stage 2 of 2” on page 193](#)
- [“Resuming the Siebel Upgrade Wizard After the Second Pause” on page 193](#)
- [“Executing SQL Output Files After the Third Pause” on page 195](#)
- [“Resuming the Siebel Upgrade Wizard After the Third Pause” on page 195](#)
- [“Updating Statistics” on page 197](#)
- [“Synchronizing the Schema” on page 198](#)
- [“Dedup Files” on page 199](#)
- [“Migrating Custom Business Component Configurations” on page 199](#)

NOTE: Be aware that when performing a development environment upgrade from the midtier, there is only a *source* system name (that is, there is no *target* system name).

Executing SQL Files After the First Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After you launch the Siebel Upgrade Wizard on the mid-tier to start the upgrade process, it generates PRET and drop files and then stops at the first pause. Before re-starting the Upgrade Wizard, execute the files that have been generated to prepare the database tables for upgrading.

Execute the files in the following order from the DB2 command window:

- **Dropifdb.sql.** Drops EIM/IF tables.

- **Drop_view.sql.** This file drops all views related to the schema.
- **Droptrig.sql.** This file drops all triggers related to the schema. This file may be empty.
- **PRET.sql.** This static file (consisting of five update statements) is run on the source tables. It performs updates to the tables, such as adding columns and revising date formats, so that unloads from the source tables can load, compatible with the target tables.

This affects source tables with columns that changed from NULLABLE to NOT NULL (not accepting null values).
- **Pret_sia.sql.** This file is only generated for upgrades of Siebel Industry Applications (SIA). It performs updates to SIA-specific tables.

To execute each SQL file

- 1 From the DB2 command window, go to the output directory that you specified (this is where the SQL files were generated) and type:

```
db2 connect to DATA_SOURCE user DATABASE_USER_NAME using PASSWORD  
db2 set current sqlid='SCHEMA_QUALIFIER'
```

The following message appears if the command is successful:

```
DB20000I The SQL command completed successfully.
```

- 2 Open the SQL file in the specified output directory and check the SQL delimiter. This can be either ";" or "/".
- 3 Execute each SQL file, in the order in which they are listed above, by entering the command that is appropriate for the SQL delimiter.

- If the delimiter is ";" then the command to execute the sql file is:

```
Db2 -stvf SQL_FILENAME.sql -z LOGFILENAME.log
```

Where:

SQL_FILENAME.sql = name of the SQL file

LOGFILENAME.log = name of the log file in which the results of this command are stored. You can specify any name.

For example, to execute drop_view.sql the command is:

```
Db2 -stvf drop_view.sql -z drop_view.log
```

- If the delimiter is "/" (or any character other than " ; "), then the command to execute the sql file is:


```
Db2 -td/ -svf SQL_FILENAME.sql -z LOG_FILENAME.log
```

Where:

SQL_FILENAME.sql = the name of the .SQL file

LOG_FILENAME.log = the name you specify for the log file in which the results of this command are stored (you can specify any name)

To execute *schema.sql* the command is:

```
Db2 -td/ -svf schema.sql -z schema.log
```

NOTE: The *-s* option in the *-stvf* command and the *-svf* command specifies that the execution of the SQL commands is to stop if an SQL error is returned. You should specify the *-s* option when running all Siebel upgrade scripts from the mid-tier. If you do not, you will have to restore your database and rerun the upgrade if you encounter errors because of data dependencies in these jobs.

- 4 Repeat [Step 3 on page 184](#) for each SQL file.

About Responding to SQL Errors

If you encounter an error when executing the SQL files, check the *errors.txt* (UNIX) or *errors.rtf* (Windows) file. If the error is listed in the error file, it is an acceptable error. In this case, delete the statements that have already executed from the SQL file and resubmit it to allow the remaining statements to execute.

If you encounter an error that is not listed in the errors file, it is an unacceptable error. You must correct the condition that caused the error before proceeding any further.

For further information, see [Chapter 14, "Reviewing the Database Upgrade Log Files."](#)

Resuming the Siebel Upgrade Wizard After the First Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

After executing the PRET and drop files as described in ["Executing SQL Files After the First Pause" on page 183](#), continue the upgrade from the point at which it paused to generate the schema and data migration files.

To continue upgrade of the Siebel Database schema on the workstation

- 1 Resume the development environment upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from [Step 4 on page 171](#), click Yes in the dialog box.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the Windows DOS command prompt from the `$SIEBEL_ROOT\BIN` directory:

```
si ebupg.exe /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

To restart a development upgrade from Release 7.0.4, type the following command:

```
si ebupg.exe /m master_upgrep_dev_704_mf.ucf
```

UNIX:

- Navigate to `$SIEBEL_ROOT/bin` and type the following command:

```
srvrupgwi z /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

To restart a development environment upgrade from Release 7.0.4, type the following command:

```
srvrupgwi z /m master_upgrep_dev_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, click OK to resume the upgrade.

A check mark appears beside each item as it is completed.

CAUTION: The upgrade process executes in a DOS command window. Do not click inside this window while the upgrade is running. If you accidentally click inside the DOS command window, the process pauses and the word *Select* precedes the title of the command window. To continue, press the space bar or escape.

The Siebel Upgrade Wizard stops again and the following message appears:

Pause #2

Please make sure your database administrator performs operations based on the files in the DDL Output Directory to complete this step. Once the operations are completed, please select Yes. To stop now and continue upgrade later, please select No. (To resume, please start the Upgrade Wizard from the command line using option /m master_upgrep_dev_<path>_mf.ucf.)

- 3 Do not click either Yes or No when the second pause message appears. Leave this dialog box open until after your DBA has executed the SQL output files as described in [“Executing SQL Output Files After the Second Pause.”](#)

Executing SQL Output Files After the Second Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The schema and data migration output files created by the Siebel Upgrade Wizard before the second pause message are generated into the default output directory or into the output directory that you specified in [Table 22 on page 164](#). You execute the output files in two stages:

■ Stage 1

In stage 1 you execute a sub-set of the output files created by the Siebel Upgrade Wizard before the second pause message. Then you deploy stored procedures and user-defined functions as described in [“Deploying Stored Procedures and User-Defined Functions” on page 188](#).

■ Stage 2

After you deploy the stored procedures and user-defined functions, you execute the remainder of the output files created by the Siebel Upgrade Wizard before the second pause message.

Windows: The default DDL output directory is `DBSRVR__ROOT\db2390\dboutput\upgrade`.

UNIX: The default DDL output directory is `DBSRVR__ROOT/db2390/dboutput/upgrade`.

CAUTION: If any of the scripts run and are not successful, data is lost and cannot be recovered. At this point, the only way to recover is to back out the changes or start over. If there are no copies of the objects, the changes cannot be backed out. Therefore, it is recommended that you make copies of your objects before running the scripts.

Executing SQL Output Files After the Second Pause - Stage 1 of 2

This section describes stage 1 of how to execute the output files created by the Siebel Upgrade Wizard before the second pause message. Execute the SQL output files in the order shown in the following procedure from the DB2 command window. See [“Executing SQL Files After the First Pause” on page 183](#) for execution procedures.

To execute the SQL output files after the second pause

1 Execute the following two files to create and update tables, including temporary tables.

- schema.sql
- tmptable.sq

2 Run the following files, as appropriate to your upgrade. These files create and populate temporary tables. Data is migrated to these tables where DML operations are performed. The data is then migrated back to the primary tables and the temporary tables are dropped.

NOTE: You may have a subset of these files, because they are produced according to upgrade path and Siebel Business application.

Run the following echannel SQL files if you are upgrading from a Siebel 6.2.1 application.

- echannel_merge_contact.sql

- echannel_merge_org_int.sql

To create and populate the temporary tables used by echannel_merge_contact.sql and echannel_merge_org_int.sql, go to the dbsrvr\db2390 directory and run the following jobs:

- eChannelAccountDivMatch.sql
- eChannelEmpContactMatch.sql

If you are a Financial Services customer, run the following jobs:

- eChannelAccountDivMatch_fins.sql
- eChannelEmpContactMatch_fins.sql.

- echannel_merge_contact_sia.sql

- echannel_merge_org_int_sia.sql

- HHMigPop.sql

NOTE: Change the HHMigPop.sql file to reflect your site-specific storage details (such as, storage group and schema qualifier). Also, since HHMigPop.sql creates objects that are required for Household_Mig_Fins.sql, make sure you run HHMigPop.sql *before* running Houshold_Mig_Fins.sql.

- Household_Mig_Fins.sql

- dev0ftp.txt

Execute this file as part of the procedure described in [“Deploying Stored Procedures and User-Defined Functions.”](#)

- ftpdev0.bat

Execute this file as part of the procedure described in [“Deploying Stored Procedures and User-Defined Functions.”](#)

After executing the SQL files successfully and reviewing the log files for unacceptable errors, you deploy the stored procedures and user-defined functions as described in [“Deploying Stored Procedures and User-Defined Functions.”](#)

Deploying Stored Procedures and User-Defined Functions

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

You must transfer and install a user-defined function (UDF) and several stored procedures onto the Siebel Database Server to support the Siebel product. Any method that transfers the necessary files to the correct location on the Database Server is acceptable.

NOTE: When deploying stored procedures and user-defined functions, you need to update the job card and job parameter information to conform to your specific standards.

Perform the following tasks to transfer the stored procedure code to the z/OS host.

Prerequisites: The following prerequisites apply to creating and deploying stored procedures:

- You must have upgraded your schema to the 1:1:1 schema structure model. This occurs when you execute the schema.sql output file (see “Executing SQL Output Files After the Second Pause - Stage 1 of 2” on page 187 for further information.)
- You must have set up your Workload Manager (WLM) environment *before* running the stored procedures.

NOTE: Setting up your WLM environment is required to install and run the UDFs and stored procedures.

- You cannot create stored procedures until *after* the second pause in the development environment upgrade process.

To transfer the stored procedure files to the z/OS host

- 1 Navigate to your DDL output directory and open the dev0ftp.txt file.

Windows: The default DDL output directory is `DBSRVR_ROOT\db2390\dboutput\upgrade`.

UNIX: The default DDL output directory is `DBSRVR_ROOT/db2390/dboutput/upgrade`.

- 2 Edit the dev0ftp.txt file using the following information:

- a Change &IP into the IP address of your DB2 host, for example ZM01.
- b Change &Username to an authorized TSO ID.
- c Change the constant SIEBELQ1 to your own dataset high-level qualifier (*DSNHLQ*), for example, SADM IN, using a command like Replace All.

Use the same high-level qualifier for all z/OS upgrade datasets. The *DSNHLQ* must be 18 characters or less and can have multiple nodes separated by periods. A node must be one to eight characters in length. The *DSNHLQ* can be the TSO ID.

- 3 After modifying the dev0ftp.txt file, save it and close the file.
- 4 Double-click ftpdev0.bat on Windows (from UNIX issue the following command: `Ftp -i > ftpdev0.bat`).

You are prompted to enter the password for the user name that you entered in dev0ftp.txt to replace &Username.

- 5 Enter the password, and then press Enter.

All files generated to this point of the upgrade process transfer from the midtier to the z/OS host.

To submit the stored procedure job on the DB2 host

- 1 After the transfer is successful, log on to the mainframe, and navigate to the *DSNHLQ*. SIEBEL.DEVO file, for example, SIEBLQ1.SIEBEL.DEVO.
- 2 Edit *DSNHLQ*. SIEBEL.DEVO, and then submit the job.

- a** In the job card set the following values:

```
CLASS=S  
MSGCLASS=X  
JOBPARAM S=YOUR_HOST_NAME
```

For example, if your Host server is QA01, the job card should look like the following:

```
//###DEVOD JOB ACCNT#, CLASS=S, MSGCLASS=X, MSGLEVEL=(1, 1),  
// TIME=1440, REGION=OM, NOTIFY=&SYSUID, LINES=9999  
//*  
/*JOBPARAM S=QA01
```

Use uppercase letters only.

- b** As indicated in the Comments section, change all references of SIEBELHLQ to your TSO ID or *DSNHLQ* using the following command:

```
C SIEBELQ1 DSNHLQ ALL
```

Where *DSNHLQ* is the dataset name high-level qualifier.

- c** On the command line, enter the following commands:

❑ C <DB2LIB1> *your_DB2_Load_DSN* ALL

For example, C <DB2LIB1> DSN710.SDSNLOAD ALL

❑ C <DB2LIB2> *your_second_DB2_Load_DSN* ALL

For example, C <DB2LIB2> DSN710.RUNLIB.LOAD ALL

❑ C &TSUBSYS *your_subsystem* ALL

For example, C &TSUBSYS Q10K ALL

❑ C &SCHEMAQUALIFIER *your_schema_qualifier* ALL

For example, C &SCHEMAQUALIFIER CQ10Knnn ALL

❑ Change the <DB2WLM> parameter to your isolated WLM environment name. This must include eight characters.

For example, if your isolated WLMENV is WLMxyz1, then you should issue the following command:

```
C <DB2WLM> 'WLMxyz1 ' ALL
```

NOTE: A trailing space was added in the preceding example to make the string eight characters long.

- d** Enter submit and then press Enter.
- e** After submitting the job, enter cancel on the command line or press PF3 to save changes.
- f** Verify that the job completed successfully:
- ❑ Verify that the RC=0.
 - ❑ Verify that dataset *DSNHLQ.DEV.JCL* was created.

3 Edit the RECEIVE member of *DSNHLQ.DEV.JCL* as appropriate, then submit the job.

a In the job card set the following values:

```
CLASS=S
MSGCLASS=X
JOBPARAM S=YOUR_HOST_NAME
```

b Comment the line that begins with //LIBS by replacing L with *.

For example, if your Host server is QA01, the job card should look like the following:

```
000001 //###SPRV   JOB ACCNT#, CLASS=S, MSGCLASS=X, MSGLEVEL=(1, 1),
000002 //           TIME=1440, REGION=OM, NOTIFY=&SYSUID, LINES=9999
000003 //*
000004 /*JOBPARAM S=QA01
000005 //*
000006 /*LIBS     JCLLIB ORDER=(DSNHLQ.JCLLIB)
000007 //*
```

c Verify that RC equals 00 to signify that the job completed successfully.

4 Execute a rexx exec to alter the received DDL library symbolics. Enter the following command from the command line:

```
TSO EX 'DSNHLQ.DEV.JCL(DDLMDR)'
```

For example, if your *DSNHLQ* is XYZ123.AA, then enter the following command after you run the RECEIVE member:

```
TSO EX 'XYZ123.AA.DEV.JCL(DDLMDR)'
```

5 Edit the IBCOPY member of *DSNHLQ.DEV.JCL* as appropriate, then submit the job.

a In the job card, set the following values:

```
CLASS=S
MSGCLASS=X
JOBPARAM S=YOUR_HOST_NAME
```

b Comment the line that begins with //LIBS by replacing L with *.

c In the line beginning //OUTPDS DD DSN=*load_module_dsname_space*

Specify the WLM environment where:

load_module_dsname_space corresponds to your isolated WLM environment.

For example, if your isolated WLM environment is WLMxyz1, then

```
DB2WLM.WLMxyz1.LOAD
```

d Verify the job card looks like the following (your Host server is QA01) and submit the job:

```
000001 //###SPRV   JOB ACCNT#, CLASS=S, MSGCLASS=X, MSGLEVEL=(1, 1),
000002 //           TIME=1440, REGION=OM, NOTIFY=&SYSUID, LINES=9999
000003 //*
000004 /*JOBPARAM S=QA01
```

```
000005 //*  
000006 //*LIBS JCLLIB ORDER=(DSNHLQ.JCLLIB)  
000007 /*
```

- e Verify that RC equals 00 to signify that the job completed successfully.
- 6 Edit the BINDSIA member of *DSNHLQ.DEV.JCL*.

- a In the job card, set the following values:

```
CLASS=S  
MSGCLASS=X  
JOBPARAM S=<YOUR HOST NAME>
```

- b Comment the line that begins with *//LIBS* by replacing *L* with ***.

- c Verify the job card looks as follows (your host server is QA01):

```
000001 //###BNDS JOB ACCNT#, CLASS=S, MSGCLASS=X, MSGLEVEL=(1, 1),  
000002 // TIME=1440, REGION=OM, NOTIFY=&SYSUID, LINES=9999,  
000003 // USER=SBLDBA  
000004 /*  
000005 /*JOBPARM S=QA01  
000006 /*  
000007 /*LIBS JCLLIB ORDER=(DSNHLQ.JCLLIB)
```

- d Execute the following change command: to replace *&SCHEMAQUALIFIER* with the target schema qualifier you are using:

```
C &SCHEMAQUALIFIER TABLEOWNER_NAME ALL
```

- e Execute the following change command, replacing *ASCEBC* with either *ASCII* or *EBCDIC* as appropriate:

- For ASCII databases, type the following command: `C 'ASCEBC' ASCII ALL`

- For EBCDIC databases, type the following command: `C 'ASCEBC' EBCDIC ALL`

- f Execute the following change command to replace *&TSUBSYS* with the subsystem you are using:

```
C &TSUBSYS SUBSYSTEM ALL
```

- g Submit the job. Verify that RC equals 00 to signify that the job completed successfully.

- h Press PF3 to save your changes.

- 7 After the stored procedure creates and binds, refresh the WLM environment. To do this, run `WLMREFSH`.

Interpreting Error Returns

The BINDSIA job sometimes fails with an RC8 with the following error in SDSF (or other job output facility) output:

```
DSNT408I SQLCODE = -454, ERROR: THE SIGNATURE PROVIDED IN THE CREATE FUNCTION  
STATEMENT FOR CQ10K006.NEXTSEQ MATCHES THE SIGNATURE OF SOME OTHER FUNCTION ALREADY  
EXISTING IN THE SCHEMA
```



```
DSNT418I  SQLSTATE      = 42723  SQLSTATE RETURN CODE
DSNT415I  SQLERRP       = DSNXCRT  SQL PROCEDURE DETECTING ERROR
```

This will cause a job status return code of 8 for CREATE and 4 for BIND. Error messages of this type are acceptable. Any other message types are unacceptable and mean the job has failed.

Executing SQL Output Files After the Second Pause - Stage 2 of 2

This section describes stage 2 of how to execute the output files created by the Siebel Upgrade Wizard before the second pause message. This section assumes that you have deployed stored procedures and user-defined functions as described in [“Deploying Stored Procedures and User-Defined Functions” on page 188](#).

To execute the SQL output files after the second pause

- Execute the SQL output files in the following order from the DB2 command window to migrate preschm, product configurator, and ISS data. These files perform DML operations: they move data between tables and change data in existing fields based on specified conditions. See [“Executing SQL Files After the First Pause” on page 183](#) for execution procedures.

NOTE: You may have a subset of these files, because they are produced according to upgrade path and Siebel Business application.

- preschm.sql
- preschm_sia.sql
- prod_configurator.sql
- prod_configurator_sia.sql
- upg_iss.sql

After executing the SQL files successfully and reviewing the log files for unacceptable errors, you may continue with the upgrade.

Resuming the Siebel Upgrade Wizard After the Second Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After you perform the task, [“Executing SQL Output Files After the Second Pause - Stage 2 of 2”](#), continue the upgrade from the point at which it paused.

To continue the upgrade of the Siebel Database schema

- 1 Resume the development environment upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from [Step 2 on page 186](#), click Yes in the dialog box for pause #2.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\BIN` directory:

```
si ebupg.exe /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

To restart a development upgrade from Release 7.0.4, type the following command:

```
si ebupg.exe /m master_upgrep_dev_704_mf.ucf
```

UNIX:

- Navigate to `$SIEBEL_ROOT/bin` and type the following command:

```
srvrupgwi z /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

To restart a development environment upgrade from Release 7.0.4, type the following command:

```
srvrupgwi z /m master_upgrep_dev_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, click OK to resume the upgrade.

A check mark appears beside each item as it is completed.

CAUTION: The upgrade process executes in a DOS command window. Do not click inside this window while the upgrade is running. If you accidentally click inside the DOS command window, the process pauses and the word *Select* precedes the title of the command window. To continue, press the space bar or escape.

The Siebel Upgrade Wizard stops again and the following message appears:

Pause #3.

Your database administrator must create Siebel Indexes using `scindx.sql` located in the Upgrade Output Directory. Once the Siebel Indexes are created, please continue by selecting Yes. To stop now and resubmit the Upgrade Wizard later, please select No. (To resume, please start the Upgrade Wizard from the command line using option `/m master_upgrep_dev_<path>_mf.ucf`.)

- 3 Do not click either Yes or No when the third pause message appears. Leave this dialog box open until after your DBA has executed the following SQL files on the z/OS host:

- `indexes.sql`
- `gen_primary1.sql`
- `gen_primary2.sql`
- `gen_primary3.sql`

- gen_primary4.sql

Executing SQL Output Files After the Third Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The output files created by the Siebel Upgrade Wizard before the third pause message are generated into the default output directory or into the output directory that you specified in [Table 22 on page 164](#).

Windows: The default DDL output directory is `DBSRVR_ROOT\db2390\dboutput\upgrade`.

UNIX: The default DDL output directory is `DBSRVR_ROOT/db2390/dboutput/upgrade`.

To execute the SQL output files after the third pause

- Execute the SQL output files in the following order from the DB2 command window. See [“Executing SQL Files After the First Pause” on page 183](#) for execution procedures.

- indexes.sql

This file creates Siebel indexes.

- gen_primary1.sql
- gen_primary2.sql
- gen_primary3.sql
- gen_primary4.sql

The gen_primary*.sql files update data in the tables.

After successfully executing the SQL files and reviewing the log files for unacceptable errors, continue with the upgrade from the point at which it paused.

Resuming the Siebel Upgrade Wizard After the Third Pause

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After your DBA has performed the task, [“Executing SQL Output Files After the Third Pause” on page 195](#) and updated statistics, you may continue the upgrade from the point at which it paused.

To continue the upgrade of the Siebel Database schema

- 1 Resume the development environment upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from [Step 2 on page 194](#), click Yes in the dialog box for pause #3.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\BIN` directory:

```
si ebupg.exe /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

To restart a development upgrade from Release 7.0.4, type the following command:

```
si ebupg.exe /m master_upgrep_dev_704_mf.ucf
```

UNIX:

- Navigate to `$SIEBEL_ROOT/bin` and type the following command:

```
srvrupgwi z /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

To restart a development environment upgrade from Release 7.0.4, type the following command:

```
srvrupgwi z /m master_upgrep_dev_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, click OK to resume the upgrade. It automatically performs the following tasks:
 - a Deletes old license keys
 - b Imports seed data
 - c Imports the repositories needed to run the repository merge (one of the current schema, two of the new schema)
 - d Upgrades the application version data to the new Siebel version

- e Updates the repository tables in preparation for the repository merge.

A check mark appears beside each item on the Upgrade Wizard screen as it is completed.

NOTE: When you are performing a development environment upgrade, the Siebel Upgrade Wizard may stop responding during the Repository Data Upgrade step. If this happens, cancel the Siebel Upgrade Wizard and from the command prompt, navigate to %Siebsrvr%\Bin and execute the RUNSTATS commands on tables S_COLUMN and S_UK_ATTJOIN using the following syntax:

```
RSTAT390 /u <database_username> /p <database_password> /c <ODBC_datasource> /d <Tableowner> /a Y /l <logfile> /T S_COLUMN
```

```
RSTAT390 /u <database_username> /p <database_password> /c <ODBC_datasource> /d <Tableowner> /a Y /l <logfile> /T S_UK_ATTJOIN
```

After executing the these commands, restart the Siebel Upgrade Wizard from the command prompt by entering `si ebugg.exe /m master_upgrep_dev_<version>.mf.ucf`.

- 3 Select Ok. You have now finished the upgrep step in the development environment upgrade.

Updating Statistics

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

This section describes how to update statistics. Perform this task on tables that receive a large number of inserts on the repository imports.

To update statistics

- 1 Navigate to the following directory:

```
SIEBEL_ROOT\bin
```

- 2 Execute the file RSTAT390.exe

The following table describes the parameters for this executable.

Argument	Description
/u	Username. For example, sbl adm. This argument is required.
/p	Password. This argument is required.
/c	ODBC Data Source. This argument is required.
/d	Siebel Table Owner. This argument is required.
/e	Specify Y to execute RUNSTATS on entire schema. Default = N.

Argument	Description
/t	Specify the table for which you want to execute RUNSTATS.
/a	Specify Y to RUNSTATS on all tables in table space. Default = N.
/i	Specify Y to ignore errors. This applies to the entire schema only. Default = N.
/l	Specify directory and filename for log file. Default filename = rstat390.log.
/s	Sample value. Default = 25.

Synchronizing the Schema

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The output files created by the Siebel Upgrade Wizard during the upgphys procedure are generated into the output directory that you specified when you ran the Database Server Configuration utility or into the default output directory (Windows: *DBSRVR_ROOT\db2390\dboutput\upgrade*; UNIX: *DBSRVR_ROOT/db2390/dboutput/upgrade*).

To synchronize the schema

- 1 Execute the SQL output files generated by the Siebel Upgrade Wizard in the following order from the DB2 command window. See [“Executing SQL Files After the First Pause” on page 183](#) for execution procedures.

NOTE: Do not use the DSNTDP2 utility to execute these files directly on the mainframe.

- Synctab.sql: Synchronizes tables
- Syncidx.sql: Synchronizes indexes
- Tmpdeduptable.sql: Creates the temporary tables used when the dedup*.sql files are executed

NOTE: (The tmpdeduptable.sq file is generated and must be executed only if you are upgrading from a pre-v7.7 release of Siebel applications.)

- Ddlview.sql: Creates the Siebel schema views based on definitions in the dbsrvr/DB2390/ddlview.sql template

- 2 After executing each SQL file, review the corresponding log file.

If you are upgrading from a v7.7 release of Siebel applications, upgphys processing is now completed.

- 3 If you are upgrading from a pre v7.7 release of Siebel applications, after successfully executing the SQL files and reviewing the log files for unacceptable errors, continue the upgrade by selecting Yes on the Siebel Upgrade Wizard screen.

Dedup Files

Upgrades: Releases 6.x, 7.0.x, 7.5.x.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

During upgphys processing, one of the last steps is generating the dedup files and dedup_prod.jcl. These files are similar to the files that you generate during the Prepare for Production Upgrade process, which is run before starting the production environment upgrade. Since both processes use the same information when generating these files, you can use the dedup files created by either the upgphys process or the Prepare for Production Upgrade process.

Migrating Custom Business Component Configurations

Upgrades: Release 6.2.1 of Siebel Financial Services applications.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Customized business component configurations (buscomps) which are based on customer extension columns or on Siebel columns in obsolete tables need to be identified and manually fixed after the upgrade. Siebel columns in obsolete tables are reconfigured and migrated during the upgrade. However, customer buscomps or custom extension columns on obsolete tables need to be migrated manually.

NOTE: Custom extension columns on tables that are upgraded (not obsolete) are retained during the upgrade.

To review a list of the buscomps that require manual migration, review the upgcust.l log file, located in the *SI/BSRVR/l*og directory.

Critical obsolete tables are listed in [Table 24](#).

Table 24. Repository Tables That Are Obsolete in Release 7

Previous Table	Suggested New Table
S_EMPLOYEE	S_CONTACT, S_USER, S_EMP_PER
S_EMP_POSTN	S_PARTY_PER
S_ORG_INT	S_ORG_EXT, S_BU
S_POSTN_RPT_REL	S_PARTY_RPT_REL

The access control buscomp migration utility is run on the Prior Customer Repository by the Siebel Upgrade Wizard during upgrade of the Siebel Database schema.

Several fields, buscomps, and columns may need to be reevaluated and recreated after your upgrade. A sample conversion script is available for migration of data to the new columns during the development upgrade.

13 Upgrading a Database from the Mainframe

This chapter describes the tasks that are specific to performing a mainframe-centric, development environment upgrade, a production test upgrade or a production upgrade. Refer to the roadmap for your upgrade in [Chapter 4, “How to Perform the Upgrade”](#) for a complete list of all the tasks you must perform to upgrade your environment to Siebel 7.8.

This chapter includes the following topics:

- [“Generating Development Environment Mainframe-Centric Upgrade Files” on page 202](#)
- [“Transferring the dedup Files” on page 203](#)
- [“Preparing the z/OS Host Environment” on page 203](#)
- [“Resuming the Siebel Upgrade Wizard After the First Pause” on page 213](#)
- [“Transferring Control Cards and Schema DDL Files to the z/OS Host” on page 215](#)
- [“Preparing the Schema and JCL Files on the z/OS Host” on page 216](#)
- [“Installing Stored Procedures on the z/OS Host” on page 219](#)
- [“Preparing Siebel-Scheduled Jobs on the z/OS Host” on page 221](#)
- [“Optimizing the Target Job Stream, Part 1” on page 224](#)
- [“Optimizing Old-Schema Index Rebuild Jobstreams” on page 225](#)
- [“Loading Log Tables on the Source Database” on page 226](#)
- [“Loading Log Tables on the Target Database” on page 227](#)
- [“Executing the Upgrade on the z/OS Host \(Phase 1 of 2\)” on page 228](#)
- [“Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause” on page 238](#)
- [“Transferring the Index Schema to the z/OS Host” on page 239](#)
- [“Preparing Siebel-Scheduled JCL” on page 240](#)
- [“Optimizing the Target Job Stream, Part 2” on page 242](#)
- [“Executing the Upgrade on the z/OS Host \(Phase 2 of 2\)” on page 243](#)
- [“Performing Intersection Table Maintenance” on page 249](#)
- [“Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause” on page 251](#)
- [“Restarting Failed Jobs on the Mainframe” on page 253](#)

Generating Development Environment Mainframe-Centric Upgrade Files

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

By default, the Siebel software contains the midtier-centric development environment upgrade files. If you want to perform a mainframe-centric development environment upgrade, you need to copy and replace these files in the DBSRVR directory to generate the appropriate upgrade output files.

To generate development environment upgrade files to run on the mainframe

- 1 Navigate to the `DBSRVR_ROOT\DB2390\390mainframedev` directory.
- 2 Run the `runme.bat` file by double-clicking on it.

The files required for the upgrade are copied to the appropriate locations as shown in [Table 25](#).

Table 25. Upgrade Output Files

Directory Name	File Name	Comments
DBSRVR\DB2390	dedup.jcl	Used for production upgrades only.
	dedup_prod.jcl	Used for production upgrades only.
DBSRVR\DB2390\UPGRADE\<PATH>\	driver_upgprep_dev_77_mf.ucf	
	driver_upgprep_dev_752_mf.ucf	
	driver_upgprep_dev_704_mf.ucf	
	driver_upgprep_dev_621_mf.ucf	SIA applications only.
	master_upgprep_dev_77_mf.ucf	
	master_upgprep_dev_752_mf.ucf	
	master_upgprep_dev_704_mf.ucf	
	master_upgprep_dev_621_mf.ucf	SIA applications only.
SI EBSRVR\ADMIN	dbsrvr.scm	Windows only.
SI EBSRVR\LOCALE\XXX where XXX= the language code of the language of the Database Configuration Wizard.	dbsrvr.scl	Windows only.
SI EBSRVR\BIN	dbsrvr_config.ksh	UNIX only.

Transferring the dedup Files

Upgrades: All upgrades.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When you run the Database Server Configuration Utility in Prepare for Production Upgrade mode, it generates a number of upgrade output files in the *DBSRVR_ROOT* directory on the Siebel Server machine from which you ran the utility. These files include the *dedup.jcl* and *dedup_prod.jcl* files. The Siebel Upgrade Wizard uses these files to remove duplicate records from the intersection tables during production test and production upgrades.

You should run the Prepare for Production Upgrade process on the same Siebel Server that you use to perform the production upgrade. If you do not, you must transfer the *dedup*.jcl* files to the *DBSRVR_ROOT\db2390* directory on the Siebel Server where the production upgrade is performed.

Preparing the z/OS Host Environment

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When you run the Siebel Upgrade Wizard on the midtier, it generates files that are stored in the midtier output directory that you specified in [Table 22 on page 164](#). When prompted by the Upgrade Wizard when it reaches Pause #1, you must transfer these files from the output directory on the midtier into datasets on the z/OS host.

Once you have transferred these files, you need to create and populate the target datasets as instructed. You will also need to modify the populated target datasets to meet your job execution requirements. To accomplish these tasks, complete the following procedures:

- [“Prerequisites for Upgrade Procedures on the z/OS Host”](#)
- [“Transferring JCL, SQL, and Binary Files to the z/OS Host” on page 204](#)
- [“Customizing the JCL UNIT Parameter Value” on page 205](#)
- [“Defining and Allocating z/OS Setup Datasets” on page 205](#)
- [“Setting Up the Upgrade Environment on the z/OS Host” on page 207](#)
- [“Preparing for Table Creation \(PRET\) on the Source Database” on page 210](#)

Prerequisites for Upgrade Procedures on the z/OS Host

The user who performs upgrade procedures on the z/OS host requires the following authorities and access:

- A thorough understanding of zSeries architecture, JCL, and TSO functions and navigation.

- A TSO account with the authorization to allocate/create datasets on the z/OS host using the high-level qualifier specified in the FTP script.
- Access to the DB2 source and target system.
- DB2 authorities to create DB2 objects and create DB2 VSAM datasets.
- Grant and Bind authority.
- DB2 Workload Manager refresh authority.

This user must have the following information:

- The dataset name high-level qualifier (*DSNHLO*) that you replaced SIEBELQ1 with in the FTP script. The *DSNHLO* must be less than or equal to 18 characters in length and may contain multiple nodes separated by periods. Each node can be between one and eight characters in length. A node more than eight characters long will cause the job to fail.
- The default JOBCLASS=Q. Make sure you use the correct job class in the generated jobcards.

Transferring JCL, SQL, and Binary Files to the z/OS Host

When the Upgrade Wizard stops at Pause # 1, you must transfer the SQL and JCL templates and the PRET files that have been generated on the midtier to the z/OS host where they can be executed.

Once the files are transferred to the mainframe, they are staging data sets and are assigned a file name with a format of VSTG*. For a list of the files transferred from the midtier and their staging data set names, see [Appendix E, "Production Upgrade Files Generated by the Upgrade Wizard."](#)

Use the following procedure to transfer files to the z/OS host.

To transfer files to the z/OS host

- 1 Navigate to your DDL output directory and open the pretftp.txt file.

Windows: The default DDL output directory is *DBSRVR_ROOT\db2390\dboutput\upgrade*.

UNIX: The default DDL output directory is *DBSRVR_ROOT/db2390/dboutput/upgrade*.

- 2 Edit the pretftp.txt file using the following information:

- a Replace &IP with the IP address or name of your z/OS host, for example ZM01.
- b Replace &Username with the your TSO ID, for example sadmi n.
- c Replace the constant SIEBELQ1 with your own dataset high-level qualifier (*DSNHLO*), for example, to your TSO ID or to a *DSNHLO* that you have authority to allocate datasets as (such as SK. SBL) using a command like Replace All.

Use the same high-level qualifier for all z/OS upgrade datasets. The *DSNHLO* must be 18 characters or less and can have multiple nodes separated by periods. A node must be one to eight characters in length. The *DSNHLO* can be the TSO ID.

- 3 After modifying the pretftp.txt file, save it, and then close the file.

- 4 Double-click pretftp.bat (from UNIX issue the following command: `Ftp -i > pretftp.bat`). You are prompted to enter the password for the user name that you entered in pretftp.txt to replace &Username.
- 5 Enter the password and then press Enter.
All files generated to this point of the upgrade process transfer from the midtier to the z/OS host.
The pretftp.log file is created in `dbsrvr\db2390\dboutput\upgrade`.

Customizing the JCL UNIT Parameter Value

If appropriate for your environment, you can amend the `UNIT=SYSDA` parameter setting for all of the JCL generated for the Siebel upgrade before you run any jobs on the mainframe. The `UNIT=SYSDA` parameter can be amended in each of the following data sets.

- `job0.txt`: dynamically allocated data sets
- `vstg0007`: `install.lib` jcl allocation
- `vstg0016`: `install.jcl` pds data sets allocations
- `vstg0200`: `pretedit` jcl
- `vstg0020`: `proclib`

To amend the UNIT parameter setting

- 1 Navigate to the data set you want to amend. Make a backup copy of the data set.
- 2 Go to Edit mode on the data set.
- 3 To change the `UNIT=SYSDA` parameter, enter the following command on the command line:

```
c sysda sgunit all
```

where *sgunit* is the unit name you want to specify.

- 4 Press PF3 to save your changes.

All the JCL generated by the data set will use the new value you specified.

NOTE: If you want to revert to the default `UNIT` parameter settings, either restore the backup copy of the data set you made in [Step 1](#) or transfer the staging data sets from the midtier again.

Defining and Allocating z/OS Setup Datasets

The following procedure defines and allocates z/OS setup datasets.

To define and allocate z/OS setup datasets

- 1 After you have successfully transferred the files generated by the Upgrade Wizard up to Pause #1, log on to the mainframe, and navigate to the *DSNHLO.SIEBEL.JOB0* dataset, for example, *SK.SBL.SIEBEL.JOB0*.

Go to Edit mode on the dataset.

- 2 Follow the procedures inside the file. Use uppercase letters only. As indicated in the Comments section, change all references of *SIEBELHLO* to *DSNHLO* using the following command:

```
c SIEBELHLO DSNHLO ALL
```

CAUTION: The variables *DSNHLO*, *SIEBELO1*, and *SIEBELHLO* all represent the same value, and this value must be the same throughout your deployment. Ensure that your job card follows your organization's naming standards.

- 3 Submit the job using the JCL in dataset *DSNHLO.SIEBEL.JOB0*.
- 4 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 5 Verify that the job ran successfully.
 - a Verify that the RC=0.
 - b Verify that three datasets were created:
 - ❑ *DSNHLO.SIEBEL.EXEC*
 - ❑ *DSNHLO.SIEBEL.JCLLIB*
 - ❑ *DSNHLO.SIEBEL.SP.CNTL*
 - c Verify that the following PDS members contain information that you defined in the Database Server Configuration Wizard on the midtier:
 - ❑ *DSNHLO.SIEBEL.EXEC(@TBOSRC)* contains the value for the source schema qualifier.
 - ❑ *DSNHLO.SIEBEL.EXEC(@TBOTAR)* contains the value for the target schema qualifier.
- 6 Enter 6 on the TSO command line to go to the TSO Command Line Processor panel.

- 7 Enter the following command to allocate the CLIST/EXEC library *DSNHLO.SIEBEL.EXEC*:

```
al loc f(sysexec) da(' DSNHLO.SIEBEL.EXEC' ) shr reuse
```

NOTE: To make sure that only Siebel executables are referenced, this command disables all CLIST libraries allocated to the logged-in user.

No message appears when the library is allocated. To verify that the exec library is listed in the file allocations by name, enter the TSO command `LSRDDN`.

Execute the `SBLUPG78` member from the *DSNHLO.SIEBEL.EXEC* REXX library to display the Siebel Upgrade Menu; you can use this menu to navigate through the steps in the upgrade process.

The EXEC library *DSNHLO.SIEBEL.EXEC* is used to execute all REXX execs. The panel and programs are located in this library.

TIP: The TSO command allocation for the EXEC library must be reissued with each TSO logon. Add this allocation to your personal logon CLIST to avoid having to reallocate the *DSNHLO.SIEBEL.EXEC* library with each new TSO logon.

Setting Up the Upgrade Environment on the z/OS Host

Once the files transferred from the midtier have been defined and allocated, you must receive the files (uncompress the files), set up the z/OS system environment and unpack the staging data sets into PDS format and create JCL members.

Perform the following steps to set up the upgrade environment on the z/OS host.

To set up the upgrade environment on the z/OS host

- 1 Enter the following command:

```
TSO SBLUPG78
```

The Siebel Upgrade 7.8 Main Menu appears. The panel ID is `SBLUPG7P`. You can find the panel ID in the bottom right corner of the screen.

TIP: To access Help on any screen, press `PF1`. To exit Help and return to the prior screen, press `PF3`.

- 2 On the Siebel Upgrade Main Menu, select option 0: RECEIVE - LOAD LIBRARY AND STORED PROCEDURE XMIT FILES.

- 3 Submit the displayed job.

NOTE: The displayed job depends on your language and upgrade path.

- 4 After submitting the job, enter `cancel` on the command line or press `PF3` to save changes.

This job receives XMIT format files. Three PDS datasets are allocated and populated with members. The three PDS dataset names are:

- *DSNHLO.SIEBEL.LOAD*

- *DSNHLO*. SI EBEL. SP. SPDDL
- *DSNHLO*. SI EBEL. SP. DBRMLI B

5 Verify that the job ran successfully.

- a Review the output in SDSF or another job output facility. Verify that the RC=0 and that the JCLTEST return code=FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.
- b Review the output to verify that all three datasets (and members) were received properly. You will get a message for each member successfully received for each of the three datasets.

From this point on, all jobs contain one of the following JCL INCLUDE members to check job step return codes. If condition codes are not met, the job abends (ends abnormally) with User 99. Acceptable return codes for each step of each job is controlled by the following three JCL test condition checks:

- JCLTEST requires RC<=4
- JCLTEST0 requires that RC=0
- JCLTEST8 requires that RC<=8

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

6 On the Siebel Upgrade 7.8 Main Menu, select option 1: SETUP - ENVIRONMENT AND DATASET SETUP.

The Environment/Dataset menu appears. The panel ID is SBLUTLP.

7 On the Environment/Dataset menu, select option 0: SET SYSTEM ENVIRONMENT VARIABLES.

The System Variables Definitions screen appears. The panel ID is SBLSETVP.

8 On the System Variables menu (Option 0), enter the following information and then press Enter:

- **DSN high level qualifier name.** The DSN high level qualifier name is displayed automatically.

NOTE: If the *DSNHLO* specified in [Step 2 on page 204 \(job0\)](#) is greater than 18 characters, you must rerun job0 using a *DSNHLO* less than or equal to 18 characters.

- **Item A.** The HOST/LPAR name is displayed automatically.
- **Item B.** Enter the DB2 WLM name, for example, DB27WLM.
- **Item C.** Enter the DB2 CCSID (encoding) type:

- A = ASCII
- E = EBCDIC

The DB2 CCSID variable is used to modify the bind jobs which are stored in *DSNHLO*. SI EBEL. SP. CNTL and the WLM parameter is updated in this dataset.

- **Item D.** The DB2 SUBSYS for source and target subsystems appears automatically.

- **Item E.** Enter the DB2 load libraries for the source and target subsystems. (These are the libraries where the DSN, DSNTEP2, and DSNTIAUL programs are located.)

NOTE: The libraries you enter must exist (that is, they must be cataloged).

- 9 Press Enter.

Messages appear indicating that the *DSNHLO* and DB2 load library information was written to individual PDS members in the *DSNHLO.SIEBEL.JCLLIB* library.

- 10 Press Enter when you see the message: . . . PRESS ENTER TO CONTINUE. . .

Press Enter again and the *JOBNAME PREFIX/PARM DEFINITIONS* panel appears. The panel ID is *SBLJPFXP*.

- 11 You must enter a three-character job name prefix for all upgrade job types for items 1 through 17.

The three-character prefix should be unique— this makes it easier to find your jobs in the queue— but it is only required for the non-unique index and obsolete index job prefixes (which cannot be the same). The remaining five characters of the job name (which do not appear and cannot be modified) are defined by Siebel Systems and are unique across all upgrade jobs.

- 12 You can change the NOTIFY value from *&SYSUID* to your TSO ID or leave it as *&SYSUID*.

NOTE: If you want to remove the notify parameter from the job card, replace the symbolic parameter *&sysuid* with spaces.

- 13 Review the job card parameters and make any necessary changes.

Verify that you are using the correct accounting, job class, and message class.

- 14 To cancel panel *SBLJPFXP* and return to the Environment/Dataset menu, press PF3.

- 15 Press Enter after entering the job name prefix and parameter definitions.

The JCL template files are updated. Messages indicate that the DB2 WLM was set and that all references to *SIEBELQ1* (midtier parameter in JCL template files) were replaced by the *DSNHLO* that you specified in [Step 2 on page 204](#).

The final message indicates that upgrade system environment variables have been set.

- 16 Press Enter to return to the Environment/Dataset menu.

- 17 On the Environment/Dataset menu, select option 1: CREATE UPGRADE JCL LIBRARY (*VSTG00nn*).

You are in edit mode for dataset *DSNHLO.SIEBEL.VSTG00nn*.

NOTE: If you want to change the job card, do so at this time.

- 18 Run the job using the JCL in dataset *DSNHLO.SIEBEL.VSTG00nn*.

- 19 Verify that the job ran successfully.

- a Review the output in SDSF or another job output facility. Verify that the RC=0 and that the JCLTEST return code=FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.

- b Verify that this job allocated and populated dataset *DSNHLO.SIEBEL.INSTALL.JCL*.

- 20 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 21 On the Environment/Dataset menu, select option 2: ALLOCATE PDS DATASETS (CREATEDS).
- 22 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(CREATEDS)*.
- 23 Verify that the job ran successfully.
 - a Review the output in SDSF or another job output facility. Verify that the RC=0 and that the JCLTEST return code=FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
 - b Verify that this job allocated datasets.
- 24 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 25 On the Environment/Dataset menu, select option 3: BUILD/POPULATE PDS DATASETS(UNPACK01).
- 26 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(UNPACK01)*.
- 27 Verify that the job ran successfully.
 - a Review the output in SDSF or another job output facility. Verify that the RC=0. A RC=4 is acceptable if the dataset is empty.
 - b Verify that the JCLTEST return code is FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
 - c Verify that this job populates (unpacks) all PDS members into corresponding PDS datasets and sequential files. It is acceptable to have some empty datasets.

NOTE: PDS datasets are populated using IEBUPDTE; sequential files are populated using IEBGENER.
- 28 Press PF3 to return to the Siebel Upgrade Main Menu.

Preparing for Table Creation (PRET) on the Source Database

The PRET (pre-table) jobs are run against the source tables prior to the beginning of the upgrade. There are two types of PRET jobs: jobs that you submit manually and jobs that run automatically. The manual PRET jobs perform the following tasks (this list varies according to your upgrade path):

- Determine whether there are clobbs in the schema
- Query catalog tables for clustering index information
- Deletes all rows from the S_DOCK_TXN_LOG and S_ESCL_REQ tables.

The automatic jobs drop some indexes, rename table(s) and add columns to tables. If you are using Siebel Scheduling, the automatic jobs use the unique job name prefixes you specify in [“Adding Jobcards to All PRET Jobs” on page 211](#).

A small number of source tables are altered by the PRET step—new columns are added— but user data in the source database remains untouched. You might, however, want to run back-up (unload) jobs on the source tables that will be altered by the PRET jobs.

Complete the following procedures to prepare the source database to generate unload, load, and schema files:

- [“Accessing the PRET Upgrade Processes Menu”](#)
- [“Adding Jobcards to All PRET Jobs”](#)
- [“Generating and Dropping Interface Tables” on page 212](#)
- [“Running Table Edit Maintenance Jobs” on page 212](#)
- [“Running Table Creation Maintenance Jobs” on page 213](#)

Accessing the PRET Upgrade Processes Menu

Complete the following procedure to access the PRET menu from where you can choose options that will modify the source database.

Navigate to the PRET menu

- 1 On the Siebel Upgrade Main Menu, select option 2: PRET UPGRADE PROCESSES.
The PRET Upgrade Processes menu appears. The panel ID is SBLPRETP.
- 2 Enter the appropriate number for the PRET job you want to perform and press Enter.
To return to the PRET Upgrade Processes menu, press PF3.

Adding Jobcards to All PRET Jobs

Perform the following procedure to add job cards to all PRET jobs and to choose your preferred job scheduling method (Siebel scheduled or vendor scheduled). Changing the job name prefixes to a unique value makes it easier to identify your jobs in the queue.

Ensure that your job card follows your organization’s standards.

To add jobcards to all jobs

- 1 On the PRET Upgrade Process menu, select option 1: ADD JOBCARDS TO PRET JOB TYPES.
- 2 When you are prompted to enter the type of scheduling for the PRET processes, enter 1: Siebel Scheduled or 2: Vendor Scheduled. For information about choosing a scheduler, see [“Executions of Jobs Using Siebel-Scheduled Mode or Vendor-Scheduled Mode” on page 91](#).

CAUTION: Choose your scheduling mode carefully. Once you select a a scheduling mode for PRET, you cannot change it.

This option builds job cards for the PRET and pretfins job types. The pretfins jobcards are only built if you are upgrading a Siebel Industry application. If you executed Household scripts on the midtier and transferred those files to the z/OS host, household jobcards are also built (no messages are displayed).

During the Schema/JCL (JCL-Prep) phase you will again be prompted to select Siebel- or Vendor-scheduled Mode. You can select a different scheduling mode than what you selected during the PRET process.

Generating and Dropping Interface Tables

Perform the following procedure to generate and drop interface tables from the source database.

NOTE: Perform this procedure only during the production environment upgrade. For information on generating and dropping interface tables during a development environment upgrade, see “Preparing EIM Tables for Upgrade” on page 153.

To generate and drop interface tables

- 1 On the Siebel Upgrade Main Menu, enter 2 to generate and drop interface tables from the source system.
This places you in edit mode on JCL member PRETI DRP for dataset *DSNHLQ.SI EBEL. I NSTALL. JCL.*
- 2 Enter sub on the command line and press Enter to submit the job using the JCL in dataset *DSNHLQ.SI EBEL. I NSTALL. JCL(PRETI DRP).*
- 3 After submitting the job, review the output in SDSF or another job output facility to verify that the job ran successfully.
 - a Verify that all steps in this job contain an acceptable return code:
 - ❑ RC=0 or RC=4 indicates the job was successful
 - ❑ RC=98 indicates that this job has already been run, so there is nothing to drop. RC=98 may be acceptable, but you should verify that all EIM interface tables do not exist under the source tableowner.
 - b Verify that the JCLTEST return code is FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
- 4 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Running Table Edit Maintenance Jobs

Perform the following procedure to run the table edit maintenance jobs called PRETEDIT.

To run PRETEDIT jobs

- 1 On the Siebel Upgrade Main Menu, select option 3: PRETEDIT JOBS. This places you in edit mode for the PDS dataset *DSNHLQ.SI EBEL. PRETEDI T. JCL.*
NOTE: Jobs from this step can run in parallel.
- 2 Submit each PDS member job using the JCL in dataset *DSNHLQ.SI EBEL. PRETEDI T. JCL.*
- 3 After all jobs are submitted, review the output in SDSF or another job output facility to verify that the job ran successfully.

- All steps must contain an acceptable return code: RC=0 or RC=4.
 - The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.
- 4 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Running Table Creation Maintenance Jobs

Perform the following procedure to run the table creation maintenance jobs called PRET.

To run the table creation maintenance jobs

- 1 On the Siebel PRET menu, select option 4: PRET - PRET JOBS.

This places you in edit mode for the PDS dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBPRET)`.

NOTE: If you are upgrading a Siebel Financial Services (FINS-SIA) application, you must first complete option 4: PRET - PRET JOBS. Once you have completed this option, you need to select option 5: PRETFINS - PRETFINS JOBS. This option places you in edit mode for the PDS dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBPRETF)`.

This job submits the first PRET job. If the first job is successful, it submits the dependent jobs. This process continues until *all* PRET jobs are run.

CAUTION: Each job must run serially due to dependencies. If a single job fails, the process stops until the failed job is corrected and run successfully.

- 2 After *all* jobs have run, review the output in SDSF or another job output facility to verify that the job ran successfully.

All steps must contain an acceptable return code: RC=0 or RC=8.

The JCLTEST or JCLTEST8 return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.

- 3 After submitting the job, enter cancel on the command line or press PF3 to save changes.

NOTE: After successfully completing this task, continue with the upgrade on the midtier.

Resuming the Siebel Upgrade Wizard After the First Pause

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

After your DBA has applied the PRET processes on the z/OS host (see ["Preparing the z/OS Host Environment" on page 203](#)), continue the upgrade from the point at which it paused, as described in the following procedure.

To continue upgrade of the Siebel database schema on the midtier

- 1 Resume the upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from Pause #1 (Step 4 on page 171), click Yes.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the DOS command prompt from the `SIEBEL_ROOT\BIN` directory:

```
si ebupg.exe /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
si ebupg.exe /m master_upgrep_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
si ebupg.exe /m master_upgrep_prod_704_mf.ucf
```

UNIX:

- Navigate to `$SIEBEL_ROOT/bin` and enter the following command:

```
srvrupgwi z /m master_UPGRADEOPTI ON_ENVI RONMENT_VERSI ON_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgwi z /m master_upgrep_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgwi z /m master_upgrep_prod_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, click OK to resume the upgrade.

A check mark appears beside each item as it is completed.

CAUTION: The upgrade process executes in a DOS command window. Do not click inside this window while the upgrade is running. If you accidentally click inside the DOS command window, the process pauses and the word *Select* precedes the title of the command window. To continue, press the space bar or Escape.

The Siebel Upgrade Wizard stops again and the following message appears:

Pause #2

Please make sure your database administrator performs operations based on the files in the DDL Output Directory to complete this step. Once the operations are completed, please select Yes. To stop now and continue upgrade later, please select No. (To resume, please start the Upgrade Wizard from the command line using option `/m master_upgrep_<env>_<path>.mf.ucf`.)

- 3 Do not click either Yes or No when the second pause message appears. Leave this dialog box open until after your DBA has performed the following tasks on the z/OS host.
After the DBA successfully performs the tasks on the z/OS host, continue with the upgrade.

Transferring Control Cards and Schema DDL Files to the z/OS Host

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Use the following procedure to transfer the unload and load control cards and schema DDL files generated by the Upgrade Wizard on the midtier up to Pause #2 to the z/OS host.

To transfer files to the z/OS host

- 1 Navigate to your DDL output directory and open the ftpfile.txt file.
Windows: The default DDL output directory is `DBSRVR_ROOT\db2390\dboutput\upgrade`.
UNIX: The default DDL output directory is `DBSRVR_ROOT/db2390/dboutput/upgrade`.
- 2 Edit the ftpfile.txt file using the following information:
 - a Replace &IP with the IP address or name of your z/OS host, for example ZM01.
 - b Replace &Username with the your TSO ID, for example sadmi n.
 - c Replace the variable SI EBELQ1 with the dataset high-level qualifier (*DSNHLO*) that you used for all upgrade datasets.
CAUTION: This *DSNHLO* must match the *DSNHLO* specified in your FTP script in “[Transferring JCL, SQL, and Binary Files to the z/OS Host](#)” on page 204.
You must use the same high-level qualifier for all z/OS upgrade datasets. The *DSNHLO* must be 18 characters or less and can have multiple nodes separated by periods. A node must be one to eight characters in length. The *DSNHLO* can be the TSO ID.
- 3 After modifying the ftpfile.txt file, save it, then close the file.
- 4 To run the FTP script, locate and double-click the batch file ftpfile.bat (from UNIX issue the following command: `Ftp -i > ftpfi le.bat`).
- 5 At the prompt, enter the password for the TSO ID user name that you entered in ftpfile.txt, and then press Enter.
All files generated to this point of the upgrade process transfer from the midtier to the z/OS host.
- 6 The ftpfile.log file is created in `DBSRVR_ROOT\db2390\dboutput\upgrade`.

Preparing the Schema and JCL Files on the z/OS Host

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Perform the following procedures to prepare the schema and JCL files on the z/OS host:

- [“Applying DSNHLO to FTP DDL, Unload Cards, and Load Cards”](#)
- [“Unpacking the Unload and Load Control Cards and Executing the DDL” on page 217](#)

Applying *DSNHLO* to FTP DDL, Unload Cards, and Load Cards

Perform the following task to apply the dataset name high-level qualifier (*DSNHLO*) to your FTP DDL, unload cards, and load cards.

To apply DSNHLO to DDL and unload/load files

- 1 On the Siebel Upgrade Main Menu, select option 3: SCHEMA/JCL - BUILD DDL AND JCL-PREP FOR UPGRADE.

The SCHEMA/JCL MENU appears. The panel ID is SBLUPGP.

- 2 On the Schema/JCL Menu, select option 0 (CHANGE UNLOAD/LOAD JCL TEMPLATES - SET DSNHLO), and then press Enter.

NOTE: This option must be run *first* and can be run only *once*. If you attempt to rerun this option, the following message appears: CHANGE UNLOAD/LOAD JCL TEMPLATES - SET *DSNHLO* HAS ALREADY BEEN RUN/APPLIED. OPTION 0 CANNOT BE RERUN.

The message STAGING dataset/TEMPLATE MODIFICATIONS IN PROGRESS appears.

Additional messages that appear:

- DROP STATEMENTS WILL BE BUILT FOR OPTIONAL USE:
 - BUILDING - DROP DATABASE STATEMENTS
 - BUILDING - DROP TABLESPACE STATEMENTS
 - BUILDING - DROP TABLE STATEMENTS
 - BUILDING - DROP INDEX (UNIQUE ONLY) STATEMENTS
 - BUILDING - DROP INDEX (NON-UNIQUE ONLY) STATEMENTS
 - BUILDING - DROP INDEX (OBSOLETE ONLY) STATEMENTS

- UNLOAD/LOAD DEFAULT *DSNHLO* SUCCESSFULLY CHANGED TO: *DSNHLO* ; PROCEED TO THE PDS/SCHEMA/VERIFICATION UTILITIES PANEL; RUN OPTION 1: BUILD/POPULATE UNLOAD/LOAD PDS DATASETS; (UNPACK02) ; MESSAGE: SBLSETV-0007

- 3 Press Enter to return to the Schema/JCL Menu.

The Schema/JCL Menu appears.

Unpacking the Unload and Load Control Cards and Executing the DDL

Complete the following procedure to unpack the unload and load cards and execute the DDL.

To unpack, unload/load, and execute the DDL

- 1 On the Siebel Upgrade Main Menu, select option 3: SCHEMA/JCL - BUILD DDL AND JCL-PREP FOR UPGRADE.

The SCHEMA/JCL MENU appears. The panel ID is SBLUPGP.

- 2 On the Schema/JCL Menu, select option 1: PDS/SCHEMA/VERIFICATION UTILITIES.

The Utilities Menu appears.

- 3 On the Utilities Menu, select option 0: BUILD/POPULATE UNLOAD/LOAD PDS DATASETS (UNPACK02).

This places you in edit mode for PDS dataset member: *DSNHLO.SIEBEL.INSTALL.JCL(UNPACK02)*.

- 4 Run the job using the JCL in dataset: *DSNHLO.SIEBEL.INSTALL.JCL(UNPACK02)*.

This job populates (unpacks, using IEBUPDTE) the unload and load control cards into members in their corresponding PDS dataset. The DDL files are output (using IEBGENER) to sequential files.

- 5 Verify that the job ran successfully, RC=0.

The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

- 7 After the batch unpack02 (option 0) job has run successfully, select option 1: CUSTOMIZE UNLOAD/LOAD FOR TABLES W/CLOB COLUMNS (CLOBCOPY).

If the source system has no CLOB columns on any tables, one of two message types appears:

- A message box that indicates that no CLOB tables exist.
- UTLO033: CLOB CUSTOMIZATION NOT REQUIRED FOR *upgtype* UPGRADE PATH.

If CLOBs exist on the source system, a series of messages appear. Read them carefully. This option will run in foreground and then place you in edit mode in the following dataset:

DSNHLO.SIEBEL.INSTALL.JCL(CLOBCOPY)

- 8 Submit the job and verify that it ran successfully, RC=0.
- 9 On the Utilities Menu, select option 2: CREATE DB2 SCHEMA (DB, TABLESPACE, TABLE, UNIQUE INDEXES, VIEW).
This places you in edit mode for PDS dataset member: *DSNHLO.SIEBEL.INSTALL.JCL(SCHEMA01)*.

NOTE: You need to verify the DDL before running the following job.

- 10 Run the job using the JCL in dataset: *DSNHLO.SIEBEL.INSTALL.JCL(SCHEMA01)*.
- 11 Verify that the job ran successfully, RC=0.
- 12 After submitting the job, enter cancel on the command line or press PF3 to save changes.

NOTE: Verify that the DDL creation (the job submitted in Step 9) was successful before continuing with the upgrade.

- 13 Select option 3: CREATE TEMP/WORK TABLE SCHEMA (3 SETS: TMPTAB00, 01 AND 02).

A message that describes the process for creating the Temp-Tables appears. This process involves being placed in edit mode three times on different PDS TMPTAB nn members in the INSTALL.JCL library:

- TMPTAB00
- TMPTAB01
- TMPTAB02

You receive an edit-mode message each time this process occurs. After accepting the message, run a batch job.

The messages have the following format:

1. YOU WILL NOW BE PLACED IN EDIT MODE FOR TMPTAB00 . . . PRESS ENTER TO CONTINUE.

NOTE: The SIEBTMP1 database in *DSNHLO.SIEBEL.TMPTAB** is a temporary database. You should be authorized to create this database. You can rename it in these members in accordance with your organization's naming standards.

- 14 At the edit-mode message for TMPTAB00, press Enter.

This places you in edit mode for PDS dataset and member *DSNHLO.SIEBEL.INSTALL.JCL(TMPTAB00)*.

- a Run this job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(TMPTAB00)*.

This job creates COMMON temp tables used by upgrade data migration SQL scripts.

- b Verify that the job ran successfully, RC=0.

The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.

- c After submitting the job, press PF3 to save changes.

- 15 At the edit-mode message for TMPTAB01, do the following:

- a** Press Enter.
This places you in edit mode for PDS dataset and member:
`DSNHLO.SI EBEL. I NSTALL. JCL(TMPTAB01)`.
 - b** Run this job using the JCL in dataset `DSNHLO.SI EBEL. I NSTALL. JCL(TMPTAB01)`.
This job creates a log table used by the upgrade process to log Unload jobs on the source subsystem. This table is used *only* if you are using Siebel Logging and Scheduling.
 - c** Verify that the job ran successfully, RC=0.
The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
 - d** After submitting the job, press PF3 to save changes.
- 16** At the edit-mode message for TMPTAB02, do the following:
- a** Press Enter.
This places you in edit mode for PDS dataset and member:
`DSNHLO.SI EBEL. I NSTALL. JCL(TMPTAB02)`.
 - b** Run this job using the JCL in dataset `DSNHLO.SI EBEL. I NSTALL. JCL(TMPTAB02)`.
This job creates a log table used by the upgrade process to log selected target subsystem jobs. This table is used *only* if you are using Siebel Logging and Scheduling.
 - c** Verify that the job ran successfully, RC=0.
The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
 - d** After submitting the job, press PF3 to save changes.

Installing Stored Procedures on the z/OS Host

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

Perform the following task to unpack the unload and load cards and execute the DDL.

To install and verify stored procedures

- 1** On the Siebel Upgrade Main Menu, select option 3: SCHEMA/JCL - BUILD DDL AND JCL-PREP FOR UPGRADE.

The SCHEMA/JCL MENU appears. The panel ID is SBLUPGP.

- 2 On the Schema/JCL Menu, select option 1: PDS/SCHEMA/VERIFICATION UTILITIES.
The Utilities Menu appears. The panel ID is SBLUTL2P.
- 3 On the Utilities Menu, select option 4: INSTALL/DEPLOY STORED PROCEDURES AND UDF.
This places you in edit mode for PDS dataset *DSNHLO.SI EBEL. SP. CNTL*.
- 4 Select the *DSNHLO.SI EBEL. SP. CNTL(@README)* PDS member.
This places you in edit mode for PDS dataset and member *DSNHLO.SI EBEL. SP. CNTL(@README)*.
- 5 Review the @README file.
- 6 Select the IEBCOPY PDS member.
This places you in edit mode for PDS dataset and member *DSNHLO.SI EBEL. SP. CNTL(IEBCOPY)*.
- 7 Follow the procedures in the JCL.
The WLM library to be used will vary based on the DB2 subsystem being used. You must replace *LOAD_MODULE_DSNAME_SPAS* with your WLM load library name.
This job moves Stored Procedure load modules to the WLMSPAS.
NOTE: Do *not* change the @DB2WLM member. This value was set on the SBLSETV panel as part of the environment/dataset setup process.
- 8 Verify that the job ran successfully, RC=0.
The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 9 Select the PDS member for your upgrade path.
 - Use BINDHOR if you are upgrading a Siebel 7.0.4, 7.5.2, or 7.7 application.
 - Use BINDSIA if you are upgrading a SIA 6.2.1, SIA/FINS/SIS 7.0.4, SIA 7.5.2, or SIA 7.7 application.This places you in edit mode for PDS dataset member *DSNHLO.SI EBEL. SP. CNTL(BINDSIA)* or *DSNHLO.SI EBEL. SP. CNTL(BINDHOR)*.
This job binds stored procedure packages.
NOTE: A DROP procedure job is included (*DSNHLO.SI EBEL. SP. CNTL(DRPSIA)* or *DSNHLO.SI EBEL. SP. CNTL(DRPHOR)*). If the procedure already exists, you must drop it before running the bind job.
- 10 Verify that the job ran successfully, RC=0 or RC=4.
The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.
- 11 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 12 Select option 5: RUN STORED PROCEDURE INSTALL VERIFICATION.
This places you in edit mode for one of the following PDS datasets and members:

- For Siebel Business application upgrades: *DSNHLO.SIEBEL.INSTALL.JCL(SPVHOR)*
- For Siebel Industry application upgrades: *DSNHLO.SIEBEL.INSTALL.JCL(SPVSI A)*

13 Run the required JCL, such as in dataset *DSNHLO.SIEBEL.INSTALL.JCL(SPVSI A)*.

This job executes each stored procedure against an empty target schema. This process is to verify that the stored procedures have been installed and can be executed.

14 Verify that the job ran successfully, RC=0.

The JCLTEST return code should be FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.

15 After submitting the job, enter cancel on the command line or press PF3 to save changes.

16 On the Utilities Menu, Press PF3.

The Schema/JCL Menu for your upgrade path appears. The panel ID is SBLUPGP.

You install the stored procedures and UDFs to facilitate upgrade processing. You can review the SQL for these objects in the dataset *DSNHQL.SIEBEL.SP.SPDDL*.

Preparing Siebel-Scheduled Jobs on the z/OS Host

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

NOTE: The procedure in this section uses Siebel-Scheduled job execution but you can also choose to use a third party job scheduler. For information on choosing a scheduler, see [“Executions of Jobs Using Siebel-Scheduled Mode or Vendor-Scheduled Mode” on page 91](#). Choose your scheduling method carefully, because once you begin an upgrade process under a selected mode, you cannot change your scheduling mode or reverse this decision.

Perform the procedures in this section to prepare the upgrade JCL and to generate non-unique index rebuilds. You can choose to generate index rebuilds in either TSO foreground mode or batch mode. It is recommended that you perform the procedure in batch mode.

To complete Siebel-scheduled JCL preparation (Part 1)

1 On the Siebel Upgrade Main Menu, select option 3: SCHEMA/JCL - BUILD DDL AND JCL-PREP FOR UPGRADE.

The SCHEMA/JCL MENU appears. The panel ID is SBLUPGP.

2 On the Schema/JCL Menu, select option 2: SIEBEL SCHEDULED - JCL-PREP TO BUILD/EXECUTE/ MANAGE JOBS VIA SIEBEL.

The JCL-PREP: Siebel Logging/Scheduling Menu appears. The panel ID is SBLJCL1P.

3 Select option 1: ADD JOBCARDS, BUILD SIEBEL JOB LOGGING CAPABILITIES.

A message appears before each part of the JCL preparation runs. The following messages are examples of the messages you receive.

NOTE: The messages you receive depend on your upgrade path.

```
SIEBEL JCL MODIFICATIONS TO BE APPLIED BASED ON RULES THAT SIEBEL LOGGING/SCHEDULING
WILL BE USED. MESSAGE: SBLJCL-0012.
```

```
JOB LOG LOADFILE BEING WRITTEN TO: DSN= DSNHLQ.SIEBEL.JOBLOG.LOADFILE.
```

```
UNLOAD JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.DDLIMP.UNLOAD.CNTL.
```

```
LOAD JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.DDLIMP.LOAD.CNTL.
```

```
DATA MI GR-PRESCHM JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.PRESCHM.JCL.
```

```
DATA MI GR-PRDCFG JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.PRDCFG.JCL.
```

```
DATA MI GR-GEN PRIMARY JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.GENPRIM.JCL.
```

```
DATA MI GR-UPGISS JCL - ADDING JOBCARDS/LOGGING DSN= DSNHLQ.SIEBEL.UPGISS.JCL.
```

```
PROCESSING COMPLETE. . .
```

```
***
```

When three asterisks (***) appear, the process is complete.

4 Press Enter.

5 From the JCL-PREP: SIEBEL LOGGING/SCHEDULING Menu (panel ID is SBLJCL1P), select option 2: ADD SIEBEL JOB LOGGING, JCL COND-CODE CHECKS.

A series of messages displays as each part of the JCL preparation runs:

NOTE: The messages you receive depend on your upgrade path.

- SIEBEL JCL MODIFICATIONS TO BE APPLIED BASED ON RULES THAT SIEBEL LOGGING/SCHEDULING WILL BE USED. MESSAGE: SBLJCL-0012.

- MODIFY UNLOAD JCL - ADDING SIEBEL LOGGING DSN= DSNHLQ.SIEBEL.DDLIMP.UNLOAD.CNTL.

- MODIFY LOAD JCL - ADDING SIEBEL LOGGING DSN= DSNHLQ.SIEBEL.DDLIMP.LOAD.CNTL.

- MODIFY DATA MI GR-PRESCHM JCL - ADDING SIEBEL LOGGING DSN= DSNHLQ.SIEBEL.PRESCHM.JCL.

- MODIFY DATA MI GR-GEN PRIMARY JCL - ADDING SIEBEL LOGGING DSN= DSNHLQ.SIEBEL.GENPRIM.JCL.

- PROCESSING COMPLETE. . .

```
***
```

When three asterisks (***) appear, the process is complete.

6 Press Enter.

- 7 From the JCL-PREP: SIEBEL LOGGING/SCHEDULING Menu (panel ID is SBLJCL1P), select option 3: ADD JOBSTEP TO EACH DATA MIGRATION JOBSTREAM THAT AUTO-SUBMITS DEPENDENT JOB TO JES INTRDR. DEPENDENT JOB SUBMITTED ONLY WHEN PREDECESSOR JOB RAN SUCCESSFULLY.

A series of messages displays as each part of the JCL preparation runs:

NOTE: The messages you receive depend on your upgrade path.

SIEBEL JCL MODIFICATIONS TO BE APPLIED BASED ON RULES THAT SIEBEL LOGGING/SCHEDULING WILL BE USED. MESSAGE: SBLJCL-0012.

PLEASE NOTE: THE FOLLOWING CHANGES WILL BE APPLIED TO SOURCE TABLE UNLOAD AND DATA MIGRATION JCL. (EXAMPLES: PRESCHM, PRDCFG, UPGISS).

MODIFY DATA MI GR-PRESCHM JCL - ADD DEPENDENT JOB SUBMIT JCL DSN=
DSNHLQ. SIEBEL. PRESCHM. JCL.

PROCESSING COMPLETE. . .

NOTE: The type and number of messages you receive depends on your upgrade.

- 8 Press Enter. The JCL-PREP: SIEBEL LOGGING/SCHEDULING Menu appears (panel ID is SBLJCL1P).

To generate non-unique index rebuilds

- 1 From the JCL-PREP: SIEBEL LOGGING/SCHEDULING Menu (panel ID is SBLJCL1P), select option 4: GENERATE INDEX REBUILDS.

The Index Menu appears. The panel ID is SBLIDXP.

The following steps describe how to rebuild non-unique indexes using the TSO foreground mode, but either batch or foreground mode can be used.

In foreground mode, messages are displayed on the screen as the JCL-Prep progresses. This mode will lock up your session until the option is complete. Each option can take an extended period of time (more than thirty minutes), depending on the user's dispatching priority.

NOTE: It is recommended that you perform the procedure in batch mode. Before doing so, make sure you modify the ispf batch procedure, *dsnhlq.siebel.proc(ispbat)*, to your installation standards. If you perform the procedure in foreground mode, make sure your logon region size is at least 7092.

- 2 Select option 1: NON-UNIQUE INDEX.

(To run option 1: NON-UNIQUE INDEX in batch mode, enter 1 in the option field, then tab to the RUN OPTIONS 1-3 IN BATCH? N (Y/N) screen. Replace the N with a Y.)

The following message appears:

SIEBEL JCL MODIFICATIONS TO BE APPLIED BASED ON RULES THAT SIEBEL LOGGING/SCHEDULING WILL BE USED. MESSAGE: SBLJCL-0014.

- 3 When the following message appears, specify the number of indexes to be included in each rebuild job:

```
ENTER THE MAXIMUM NUMBER OF INDEXES TO BE INCLUDED IN A SINGLE REBUILD JOB. (PRESS
ENTER TO ACCEPT DEFAULT OF 3, MAXIMUM IS 10)
```

CAUTION: Consider your objective before choosing a maximum number of indexes for each job. Increasing this number results in fewer jobs but requires more memory and sort work. Reducing this number results in more jobs—reducing resource requirements but building fewer indexes in parallel.

During the Non-Unique Index Rebuilds process, the following series of message appears as the process runs.

```
GENERATING INDEX REBUILD JOBS FOR NON-UNIQUE INDEXES OLD-SCHEMA BUILDING NON-UNIQUE-
INDEX REBUILDS (DDLNUIND) DSN= DSNHLQ.SIEBEL.DDLNUIND.JCL
DSN= DSNHLQ.SIEBEL.DDLNUIND.SQL.
```

```
NUMBER OF INDEX REBUILD JOBS BUILT = 951.
```

Index count varies according to your upgrade path and the number of indexes included in each job.

```
GENERATING INDEX REBUILD JOBS FOR OBSOLETE INDEXES OLD-SCHEMA - BUILDING OBSOLETE-
INDEX REBUILDS (DDLNUIND) DSN= DSNHLQ.SIEBEL.DDLNUIND.JCL
DSN= DSNHLQ.SIEBEL.DDLNUIND.SQL.
```

```
NUMBER OF INDEX REBUILD JOBS BUILT = 101.
```

Index count varies according to your upgrade path and the number of indexes included in each job.

Optimizing the Target Job Stream, Part 1

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

To improve performance and reduce the number of jobs managed in a one-table-per-tablespace database configuration, you can use an optional feature called Job Stream Optimization (JSO) to eliminate *unload* and *load* jobs for tables that contain no data.

CAUTION: Do not perform this task if your environment has multiple tables per tablespace. This process is irreversible unless you make a copy of your datasets before deleting jobs that act on tables containing no data.

This section applies to options 0 and A on the Job-Stream Optimization menu.

To remove unload and load jobs for tables that contain no data

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select Option 4. OPTIMIZATION - JOB-STREAM OPTIMIZATION (OPTIONAL) and then press Enter.
The *** UPGRADE <upgrade type> TO 7.8 JOB-STREAM OPTIMIZATION MENU appears. The panel ID is SBLJSOP.
- 3 Select Option 0: IDENTIFY "EMPTY" TABLES (REQUIRED-MUST RUN FIRST) and then press Enter.
This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC0)*.
NOTE: This option is only available if your source database follows the 1:1:1 database schema structure.
- 4 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC0)*.
- 5 Submit the job, press Enter, and then press PF3.
NOTE: Verify the return code to make sure the job ran successfully before proceeding to the next step.
- 6 Select Option A) 1. GENERATE LIST OF UNLOAD/LOAD JOBS and then press Enter.
This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC1)*.
- 7 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC1)*.
- 8 Submit the job, press Enter, and then press PF3.
NOTE: Make sure the job ran successfully by verifying the return code before proceeding to the next step.
- 9 Select Option A) 2. REMOVE UNLOAD/LOAD JOBS and then press Enter.
This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC2)*.
- 10 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC2)*.
- 11 Submit the job, press Enter, and then press PF3.

Optimizing Old-Schema Index Rebuild Jobstreams

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

To improve performance and reduce the number of jobs run, you can use an optional feature called Job Stream Optimization (JSO) to eliminate index rebuild jobs for tables that contain no data.

This section applies to options 0 and A on the Job-Stream Optimization menu.

To optimize old-schema index rebuild jobstreams

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select Option 4. OPTIMIZATION - JOB-STREAM OPTIMIZATION (OPTIONAL) and then press Enter.
The UPGRADE <upgrade type> TO 7.8 JOB-STREAM OPTIMIZATION MENU appears.
- 3 Select Option A) 3. GENERATE LIST OF OLD-SCHEMA INDEX REBUILD JOBS and then press Enter.
This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC3)*.
- 4 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC3)*.
NOTE: After running this job, verify that it was successful by reviewing the sysout in SDSF or another job output facility and making sure the RC=0.
- 5 Submit the job, press Enter, and then press PF3.
- 6 Select Option A) 4. REMOVE OLD-SCHEMA INDEX REBUILD JOBS and press Enter.
This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC4)*.
- 7 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(JSOSRC4)*.
- 8 Submit the job, press Enter, and then press PF3 twice.
The *** UPGRADE <upgrade type> TO 7.8 MAIN MENU *** appears. The panel ID is SPLUPG7P.

Loading Log Tables on the Source Database

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

Perform the following procedure to load log tables on the source database.

To load the Source Log table

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.
The Siebel Upgrade Job Submission Menu for your upgrade path appears.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

- 3 Select option 0: LOAD SOURCE SIEBEL LOG.
This loads the source Siebel log table using the *dsnhlq.siebel.joblog.loadfile*.
- 4 Verify that the job class is appropriate for the source subsystem.
- 5 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(LSRCLOG)*.
NOTE: After running this job, verify that it was successful by reviewing the sysout in SDSF or another job output facility and making sure the RC=0.
- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Loading Log Tables on the Target Database

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

Perform the following steps to load log tables on the target database.

To load the Target Log table

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.
The Siebel Upgrade Job Submission Menu for your upgrade path appears.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 3 Select option 1: LOAD TARGET SIEBEL LOG (1 OF 2).
- 4 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(LTARLOG1)*.
This loads the target Siebel log table using the *dsnhlq.siebel.joblog.loadfile*.
NOTE: After running this job, verify that it was successful by reviewing the sysout in SDSF or another job output facility and making sure the RC=0.
- 5 Review sysout output, rows loaded will vary, depending on your upgrade path.
- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Executing the Upgrade on the z/OS Host (Phase 1 of 2)

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

To execute the upgrade process on the z/OS host, complete the following procedures:

- [“Running Unload Jobs on the Source Database”](#)
- [“Running Load Jobs on the Target Database” on page 229](#)
- [“Executing Index DDL and Executing Rebuild Indexes” on page 230](#)
- (Optional.) [“Migrating eChannel Data” on page 233](#)
- (Optional.) [“Migrating Household Data” on page 234](#)
- [“Migrating Preschm Data” on page 231](#)
- [“Migrating Address Data” on page 232](#)
- [“Migrating Product Configurator Data” on page 235](#)
- [“Migrating ISS Data” on page 237](#)

Siebel Financial Services Upgrades

Complete the following procedures if you are upgrading a Siebel Financial Services application:

- [“Migrating Preschm Data for FINS” on page 234](#)
- [“Migrating Product Configurator Data for FINS” on page 236](#)

Running Unload Jobs on the Source Database

Perform the following procedure to run unload jobs on the source database.

To run unload jobs on the source database

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.
The Siebel Upgrade Job Submission Menu for your upgrade path appears.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

- 3 Select option 2: UNLOAD SOURCE.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBUNLD)`.

- 4 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBUNLD)`.

This job automatically submits *all* the unload job streams referenced in dataset `DSNHLO.SIEBEL.EXEC(@DDLIMPU)`. The number of Unload jobs varies according to upgrade path.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the source Siebel log table: `SOURCE_TABLEOWNER.TMP_SBLLOG_SRC`.

- 5 Verify that the job ran successfully, RC=0 or RC=4.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

- 7 View the job status log.

If you selected Siebel-Scheduled Mode, you can view the job status log by completing the procedure described in [“Viewing the Siebel Job Log Status” on page 258](#).

Running Load Jobs on the Target Database

Perform the following procedure to run load jobs on the target database.

To run loads to the target database

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The Siebel Upgrade Job Submission Menu for your upgrade path appears.

- For Siebel Business application upgrades, the panel ID is SBLRUNHP.

- For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

- 3 Select option 3: LOAD TARGET.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBLOAD)`.

- 4 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBLOAD)`.

This job automatically submits *all* the unload job streams referenced in dataset `DSNHLO.SIEBEL.EXEC(@DDLIMPL)`.

This submits the Load job streams. The number of upgrade Load jobs varies by upgrade path. If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

- 5 Verify job ran successfully, RC=0 or RC=4.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe”](#) on page 253.

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 7 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status”](#) on page 258.

Once all Load jobs have completed successfully, continue with the upgrade.

Executing Index DDL and Executing Rebuild Indexes

Perform the following procedure to execute index DDL and execute rebuild indexes.

To execute the index DDL and rebuild indexes

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The Siebel Upgrade Job Submission Menu for your upgrade path appears.

- For Siebel Business application upgrades, the panel ID is SBLRUNHP.
- For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

- 3 Select option 4: "OLD SCHEMA INDEX DDL".

This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(SCHEMA02)*.

- 4 Run the job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(SCHEMA02)*.

This job runs DDL Create Index statements to build the old-schema non-unique and obsolete indexes.

- 5 Verify that the job ran successfully, RC=0 or RC=4 in steps DDLNUIND and DDLOIND.

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

The Siebel Upgrade Job Submission Menu for your upgrade path appears. The panel ID is SBLRUNFP.

- 7 Select option 5: "OLD SCHEMA INDEX REBUILDS".

This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(SUBIX)*.

- 8 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBIX)`.

This job automatically submits *all* Non-Unique and Obsolete Index Rebuild jobs referenced in datasets: `DSNHLO.SIEBEL.EXEC(@DDLNUIX)` and `DSNHLO.SIEBEL.EXEC(@DDL0IX)`.

Non-unique job names range from `###N0001` to `###Nnnnn`. Obsolete job names range from `###O0001` to `###Onnnn`, where the placeholder `###` represents your job prefix and `nnnn` represents a sequential number from 0001 to *n*.

- 9 Verify that the job ran successfully, RC=0 or RC=4.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe”](#) on page 253.

- 10 After submitting the job, enter `cancel` on the command line or press PF3 to save changes.
- 11 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status”](#) on page 258.

Migrating Preschm Data

The PRESCHM jobs migrate pre-7.8 Siebel data to version 7.8 data. Generally, this involves inserting or updating values in the target tables but new indexes may also be created and rebuilt. The PRESCH jobs also create temporary indexes for data migration.

There are optional data migration scripts for eChannel and Household data and for Siebel Financial Services (FINS) applications.

Some of the PRESCHM jobs run independently but others are submitted in a defined order and cannot run until previous jobs have completed successfully; you can run the stand-alone jobs in parallel with those with dependencies. To see the serial flow of the PRESCHM jobs and the PRESCHM job dependencies, look at the PDS member `DSNHLO.SIEBEL.PRESCHM.JCL(@TRIGGER)`.

To migrate preschm data

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The Siebel Upgrade Job Submission Menu for your upgrade path appears.

- For Siebel Business application upgrades, the panel ID is SBLRUNHP.
- For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

3 Select option 6: PRESCHM.

This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(SUBPSH)*.

4 Run the job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(SUBPSH)*.

NOTE: This job automatically submits the *first* PRESCHM job referenced in dataset *DSNHLO.SIEBEL.EXEC(@PRESCHM)*.

- If this job runs successfully, it automatically submits the next dependent job, and so on.
- This process continues for *all* PRESCHM jobs. *If a job fails, the succeeding dependent job does not get submitted.* The automatic job submission sequence terminates.

5 After *all* PRESCHM jobs are run, verify that each job ran successfully.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs that have an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: *TARGET_TABLEOWNER.TMP_SBLLOG_TAR*.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

6 Enter cancel on the command line or press PF3 to save changes.

7 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

Migrating Address Data

The Address Migration job corresponds to scripts run previously on the midtier against the Source system. This means that you *only* need to run the following job if you ran the midtier Address Migration script (*rpt_dup_addr_rowids.sql*) and applied the midtier procedures previously noted for rowids identified by this script. Otherwise, you do not need to run this job. For further information, see [“Preparing Address Data for Upgrade” on page 143](#).

Upgrades:

- Release 6.2.1 and 7.0.x of Siebel Financial Services applications to Release 7.8.x of Siebel Industry applications.
- Release 7.8.x of Siebel Business Applications to Release 7.8.x of Siebel Industry Applications.

To run address data migration SQL scripts

1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```


- 2 Select option 4: RUN UPGRADE / POST-UPGRADE MAINTENANCE.
The panel ID is SBLRUNFP.
- 3 Select option 7: ADDRESS MIGRATION.
This places you in edit mode for dataset `DSNHLQ.SIEBEL.INSTALL.JCL(ADMIGFIN)`.
- 4 Run the job using the JCL in dataset `DSNHLQ.SIEBEL.INSTALL.JCL(ADMIGFIN)`.
- 5 Submit the job and then press Enter.
- 6 Verify that the job ran successfully, RC=0 or RC=4.
Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.
- 7 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Migrating eChannel Data

Complete the following procedure to run corresponding eChannel SQL scripts against the target system.

NOTE: This topic applies to upgrades from Release FINS 6.2.1 applications. Perform this procedure *only* if you ran corresponding eChannel scripts on the source system during the development environment upgrade.

To run the data migration eChannel SQL scripts

- 1 Locate the `DSNHLQ.SIEBEL.INSTALL.JCL` library in your dataset list.
This is a PDS, so all PDS members appear.
- 2 At the command line, locate PDS members that begin with ECHM by entering the following command:

```
L ECHM
```


The following PDS members appear in the list:
 - ECHMCON
 - ECHMORG
- 3 Enter E next to the PDS members that apply to your upgrade path.
This places you in edit mode on individual PDS members.
- 4 Run this job using the JCL in dataset `DSNHLQ.SIEBEL.INSTALL.JCL(ECHM*)`.
- 5 Verify that the job ran successfully, RC=0 or RC=4.
Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

NOTE: These jobs are not logged. You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.
This places you in edit mode on the next PDS member.

Migrating Household Data

Complete the following procedure to run corresponding Household SQL scripts against the target system.

NOTE: This topic applies to upgrades from FINS 6.2.1, upgrades from SIA 7.0.4, 7.5.2 and 7.7.x, and upgrades from Release 7.x applications. To determine whether or not you need to run this procedure, check if the DSNHLQ.SIEBEL.EXEC(@HHFIN) PDS member contains a job to be submitted; if it does, you will need to run this procedure.

To run the data migration Household SQL scripts

- 1 Locate the *DSNHLQ.SIEBEL.INSTALL.JCL* library in your dataset list.
This is a PDS, so all PDS members appear.
- 2 At the command line, locate PDS members that begin with HHMIG by using the following command:

```
L HHMIGFIN
```
- 3 Edit the HHMIGFIN member of *DSNHLQ.SIEBEL.INSTALL.JCL*.
This places you in edit mode on PDS member *DSNHLQ.SIEBEL.INSTALL.JCL(HHMIGFIN)*.
- 4 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(HHMIGFIN)*.
- 5 Verify that the job ran successfully, RC=0 or RC=4.
Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.
NOTE: These jobs are not logged. You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).
- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Migrating Preschm Data for FINS

Complete the following procedure to migrate preschm data for Siebel Financial Services (FINS) applications.

Prerequisite task: [“Migrating Preschm Data” on page 231](#).

NOTE: This topic applies to upgrades from Siebel Financial Services 6.2.1, 7.0.4, 7.5.2, and 7.7 applications.

To migrate preschm data for FINS applications

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.
The panel ID is SBLRUNFP.
- 3 Select option 8: PRESCHM-FINS.
This places you in edit mode for dataset *DSNHLO.SI EBEL. I NSTALL. JCL(SUBPSF)*.
- 4 Run the job using the JCL in dataset *DSNHLO.SI EBEL. I NSTALL. JCL(SUBPSF)*. This job automatically submits the *first* PRESCHM job referenced in dataset *DSNHLO.SI EBEL. EXEC(@PRESCHF)*.
 - If this job runs successfully, it automatically submits the next dependent job, and so on.
 - This process continues for *all* PRESCHM-FINS jobs. *If a job fails, the succeeding dependent job does not get submitted.* The automatic job submission sequence terminates.
- 5 After *all* PRESCHM FINS jobs have run, verify that each job ran successfully.
RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs that have an X in position 3 of the member name.
Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.
If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: *TARGET_TABLEOWNER.TMP_SBLLOG_TAR*.
NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).
- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 7 View the job status.
For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

Migrating Product Configurator Data

Complete the following procedure to migrate Product Configurator data.

NOTE: This topic applies to upgrades from SIA 6.2.1 applications and upgrades from Siebel Financial Services (FINS) 6.2.1 and 7.0.4 (*not* SIS 7.0.4) applications.

To migrate Product Configurator data

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The panel ID is SBLRUNFP.

- 3 Select option 9: PROD CONFIG.

This places you in edit mode for dataset *DSNHLQ.SIEBEL.INSTALL.JCL(SUBPC)*.

- 4 Run the job using the JCL in dataset *DSNHLQ.SIEBEL.INSTALL.JCL(SUBPC)*.

This job automatically submits the *first* PRODCFG job stream. If this job runs successfully, it automatically submits the next dependent job, and so on. This process continues for *all* PRODCFG jobs. *If a job fails, the succeeding dependent job does not get submitted.* The automatic job submission sequence terminates.

- 5 After the job is run, verify that the job ran successfully.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs that have an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, each job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: *TARGET_TABLEOWNER.TMP_SBLLOG_TAR*.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.

- 7 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

Migrating Product Configurator Data for FINS

Complete the following procedure to migrate Product Configurator data for Siebel Financial Services (FINS) applications.

Prerequisite task: [“Migrating Product Configurator Data” on page 235](#).

NOTE: This topic applies to upgrades from Siebel Financial Services (FINS) 6.2.1 and 7.0.4 (*not* SIS 7.0.4) applications.

To migrate Product Configurator data for FINS applications

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The Siebel Upgrade Job Submission Menu for your upgrade path appears.

The panel ID is SBLRUNFP.

- 3 Select option 10: PROD CONFIG-FINS.

This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(SUBPCF)*.

- 4 Run the job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(SUBPCF)*.

This job automatically submits the *first* PRODCFGF job stream. If this job runs successfully, it automatically submits the next dependent job, and so on. This process continues for *all* PRODCFGF jobs. *If a job fails, the succeeding dependent job does not get submitted.* The automatic job submission sequence terminates.

- 5 After the job is run, verify that the job ran successfully.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs that have an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: *TARGET_TABLEOWNER.TMP_SBLLOG_TAR*.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 7 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

Migrating ISS Data

Complete the following procedure to migrate ISS data.

Prerequisite task: [“Migrating Product Configurator Data for FINS” on page 236](#)

To migrate UPGISS data

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 5: RUN UPGRADE / POST-UPGRADE MAINTENANCE.

The Siebel Upgrade Job Submission Menu for your upgrade path appears.

- For Siebel Business application upgrades, the panel ID is SBLRUNHP.

- For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.

- 3 Select the option for the application you are upgrading.

- For Siebel Business application upgrades, select option 8: UPGISS.

- For Siebel Industry application upgrades, select option 11: UPGISS.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBUPGIS)`.

- 4 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBUPGIS)`.

This job automatically submits the *first* UPGISS job stream. If this job runs successfully, it automatically submits the next dependent job, and so on. This process continues for *all* UPGISS jobs. *If a job fails, the succeeding dependent job does not get submitted.* The automatic job submission sequence terminates.

- 5 After the job is run, verify that the job ran successfully.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs that have an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 6 After submitting the job, enter cancel on the command line or press PF3 to save changes.
- 7 View the job status.

For information on how to view job status if you selected Siebel-Scheduled Mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

Resuming the Siebel Upgrade Wizard on the Midtier After the Second Pause

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Once you have completed the task, [“Executing the Upgrade on the z/OS Host \(Phase 1 of 2\)” on page 228](#), continue the upgrade from the point at which it paused.

To resume the Siebel Upgrade Wizard on the midtier after the second pause

- 1 Resume the upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from [Step 2 on page 214](#), click Yes in the dialog box for Pause #2.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\BIN` directory:

```
si ebugg.exe /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
si ebugg.exe /m master_upgprep_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
si ebugg.exe /m master_upgprep_prod_704_mf.ucf
```

UNIX:

- Navigate to \$SIEBEL_ROOT/bin and enter the following command:

```
srvrupgwi z /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgwi z /m master_upgprep_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgwi z /m master_upgprep_prod_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, click OK to resume the upgrade.

A check mark appears beside each item as it is completed.

CAUTION: The upgrade process executes in a DOS command window. Do not click inside this window while the upgrade is running. If you accidentally click inside the DOS command window, the process pauses and the word *Select* precedes the title of the command window. To continue, press the space bar or Escape.

The Siebel Upgrade Wizard stops again and the following message appears:

Pause #3

Please create Siebel Indexes using scindx.sql located in the DDL Output Directory. Once the Siebel Indexes are created, please select Yes. To stop now and continue upgrade later, please select No. (To resume, please start the Upgrade Wizard from the command line using option /m master_upgprep_<env>_<path>_mf.ucf.)

- 3 Do not click either Yes or No when the third pause message appears. Leave this dialog box open until after you have created the Siebel indexes on the z/OS host.

Transferring the Index Schema to the z/OS Host

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Use the following procedure to transfer the index schema and dedup files created by the Upgrade Wizard on the midtier to the z/OS host.

To transfer the index schema to the z/OS host

- 1 Navigate to the `DBSRVR_ROOT\DB2390\DBOUTPUT\UPGRADE` directory and open the `ftpscindx.txt` file.
- 2 Replace `&IP` with the IP address or name of your z/OS host, for example `ZM01`.
- 3 Replace `&Username` with the your TSO or ID user name, for example `SADMIN`.
- 4 Replace the constant `SIEBELQ1` with the dataset high-level qualifier (*DSNHLO*) to be used for all upgrade datasets. The *DSNHLO* must be 18 characters or less and can have multiple nodes, separated by a period. A node must be 1-8 characters in length.

The *DSNHLO* can be the TSO ID.

NOTE: The *DSNHLO* you use *must match the DSNHLO* you used in [Step 2 on page 204](#).

- 5 After modifying the `ftpscindx.txt` file, save it, and then close the file.
- 6 To run the FTP script, find and double-click the `ftpscindx.bat` file.
- 7 At the prompt, enter the password for the TSO ID userid specified in [Step 3](#), and then press Enter.
This transfers the `SCINDEX.SQL` file from the midtier to the z/OS host; its host staging dataset name is `DSNHLO.SIEBEL.VSTG0110`.
- 8 Review the `ftpscindx.log` file to verify that all files transferred successfully.

Preparing Siebel-Scheduled JCL

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Perform the following procedure to prepare the 7.8 index DDL, build index rebuild jobs, and build drop statements for old schema indexes.

NOTE: The following procedure applies *only* if you are using the Siebel-Scheduled Mode for executing jobs.

To prepare Siebel-scheduled JCL

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
`TSO SBLUPG78`
- 2 Select option 3: SCHEMA/JCL - BUILD DDL AND JCL-PREP FOR UPGRADE.
The Schema/JCL Menu appears. The panel ID is `SBLUPGP`.

- 3 Select option 2: SIEBEL SCHEDULED - JCL-PREP TO BUILD/EXECUTE/MANAGE JOBS VIA SIEBEL.
- 4 Select option 4: GENERATE INDEX REBUILDS. The Index Menu appears. The panel ID is SBLIDXP.
- 5 The example text uses the TSO foreground mode (either batch or foreground mode can be used).

NOTE: It is recommended that you perform the procedure in batch mode.

Run option 2: GEN PRIMARY INDEX REBUILDS in batch mode by entering 2 in the option field, then tabbing to the RUN OPTION 1-OR-2 IN BATCH? N (Y/N) screen. Replace the N with a Y.

- 6 Select option 2: GEN PRIMARY INDEX REBUILDS.

A series of messages are displayed beginning with the following:

```
==> SPLIT AND BUILD "SCI NDX" INDEX DDL FILES:
DROP INDEXES      : DSN=DSNHLQ. SIEBEL. SCI NDX. GPDRPIX.DDL
EIM               : DSN=DSNHLQ. SIEBEL. SCI NDX. EIM.DDL
PRE-GEN_PRIMARY  : DSN=DSNHLQ. SIEBEL. SCI NDX. GPPRE.DDL
POST-GEN_PRIMARY : DSN=DSNHLQ. SIEBEL. SCI NDX. GPPST.DDL
```

Messages continued:

```
==> JCL TO RUN DROP/CREATE INDEX "DDL" WRITTEN TO:
DROP INDEXES      : DSN=DSNHLQ. SIEBEL. INSTALL. JCL(@GPDRPIX)
CREATE INDEXES    : DSN=DSNHLQ. SIEBEL. INSTALL. JCL(@EIMIX)
DSN=DSNHLQ. SIEBEL. INSTALL. JCL(@GPPREIX)
```

The Split and Build runs next.

The following message appears:

```
NOTE: FOLLOWING THE MESSAGES BELOW IN APPROX 15 SECONDS, AN "ENTER" PROMPT WILL
APPEAR REQUIRING YOUR ENTRY...
```

These additional messages appear:

```
DROP STATEMENTS WILL BE BUILT FOR OPTIONAL USE:
BUILDING - DROP DATABASE STATEMENTS
BUILDING - DROP TABLESPACE STATEMENTS
BUILDING - DROP TABLE STATEMENTS
BUILDING - DROP INDEX (UNIQUE ONLY) STATEMENTS
BUILDING - DROP INDEX (NON-UNIQUE ONLY) STATEMENTS
BUILDING - DROP INDEX (OBSOLETE ONLY) STATEMENTS
```

- 7 At the following prompt, enter the maximum number of indexes to be built in a single job.

ENTER THE MAXIMUM NUMBER OF INDEXES TO BE INCLUDED IN A SINGLE REBUILD JOB. (PRESS ENTER TO ACCEPT DEFAULT OF 3, MAXIMUM IS 10)

CAUTION: Consider your objective before choosing a maximum number of indexes for each job. Increasing this number results in fewer jobs but requires more memory and sort work. Reducing this number results in more jobs—reducing resource requirements but building fewer indexes in parallel.

The JCL generation for the Gen Primary Index Rebuilds runs approximately 40 minutes. The following messages appear as the generation runs:

```
*** => GENERATING INDEX REBUILD JOBS FOR INDEXES TO BE REBUILT "***BEFORE***"
GEN_PRIMARY DATA MIGRATION IS RUN

"NEW-SCHEMA" - SIEBEL 7.8 BUILDING "PRE"-GEN_PRIMARY INDEX REBUILDS (GPPREIX)

DSN=DSNHLQ.SIEBEL.SCI NDX.GPPRE.JCL

DSN=DSNHLQ.SIEBEL.SCI NDX.GPPRE.SQL
```

The number of generated index rebuilds appears.

The following message appears:

```
NUMBER OF INDEX REBUILD JOBS BUILT = 2226

PROCESSING COMPLETE. . .
```

NOTE: Index count varies according to your upgrade path and the number of indexes included in each job.

- 8 Press Enter when three asterisks (***) appear.
- The Index Menu reappears. The panel ID is SBLIDXP.

Optimizing the Target Job Stream, Part 2

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

To improve performance and reduce the number of jobs managed in a one-table-per-tablespace database configuration, you can use an optional feature called Job Stream Optimization (JSO) to eliminate index rebuild jobs for tables that contain no data.

This section applies to option B (items 5 and 6) on the Job-Stream Optimization menu.

To generate primary rebuilds

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select Option 4. OPTIMIZATION - JOB-STREAM OPTIMIZATION (OPTIONAL) and then press Enter.
The UPGRADE <upgrade type> TO 7.8 JOB-STREAM OPTIMIZATION MENU appears.
NOTE: If you have not already run Option 0: IDENTIFY "EMPTY" TABLES (REQUIRED-MUST RUN FIRST), you must do so now.
- 3 Select Option B) 5. GENERATE LIST OF NEW-SCHEMA (GEN-PRIM) REBUILDS and press Enter.
This places you in edit mode for PDS dataset and member *dsnhlq.siebel.install.jcl* (JSOTAR1).
- 4 Run the job using the JCL in dataset <DSNHLQ.SIEBEL.INSTALL.JCL(JSOTAR1)>.
- 5 Submit the job and then press Enter.
- 6 Once you have verified the success of the job, press PF3.
- 7 Select Option B) 6. REMOVE NEW-SCHEMA (GEN-PRIM) INDEX REBUILDS JOBS and then press Enter.
This places you in edit mode for PDS dataset and member *dsnhlq.siebel.install.jcl* (JSOTAR2).
- 8 Run the job using the JCL in dataset <DSNHLQ.SIEBEL.INSTALL.JCL(JSOTAR2)>.
- 9 Submit job and then press Enter.
- 10 Once you have verified the success of the job, press PF3.
- 11 Review the output in SDSF or another job output facility for pre gen primary index rebuilds.

Executing the Upgrade on the z/OS Host (Phase 2 of 2)

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

To execute the upgrade process on the z/OS host, complete the following procedures:

- ["Loading Target Siebel Log Tables" on page 244](#)
- ["Dropping Old Indexes and Creating New Indexes" on page 244](#)
- ["Rebuilding the Gen Primary 7.8 Indexes" on page 246](#)
- ["Migrating the Gen Primary 7.8 SQL" on page 246](#)
- ["Creating the New EIM Indexes" on page 247](#)
- ["Generating RUNSTATS" on page 248](#)

Loading Target Siebel Log Tables

Perform the following procedure to load target Siebel log tables. The first time you perform this procedure, it loads load jobs and unload jobs. The second time you perform it, it loads logging records for the new schema indexes (GEN PRIMARY indexes).

To load target Siebel log tables

- 1 If you are not on the Siebel Upgrade Main Menu, enter the following command:
TSO SBLUPG78
- 2 Select option 5: UPGRADE - RUN UPGRADE/POST-UPGRADE MAINTENANCE.
The Siebel Upgrade Job Submission Menu appears.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 3 Select the option for the application you are upgrading.
 - For Siebel Business application upgrades, select option 9: LOAD TARGET LOG (2 of 2).
 - For Siebel Industry application (SIA) upgrades, select option 12: LOAD TARGET LOG (2 of 2).This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(LTARLOG2)*.
- 4 Run the job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(LTARLOG2)*.
This loads the target Siebel log table using logload file *DSNHLO.SIEBEL.JOBLOG.LOADFILE*.
After the job is run, verify that the job ran successfully. (RC=0 or RC=4 indicates that the job was successful.)
- 5 After submitting the job, enter cancel on the command line or press PF3 to save changes.

Dropping Old Indexes and Creating New Indexes

Perform the following procedures to drop old schema indexes and create new schema indexes.

To drop old index schemas

- 1 Make sure you are at the Siebel Upgrade Job Submission Menu for your upgrade.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 2 Select the option for the application you are upgrading.
 - For Siebel Business application upgrades, select option 10: OLD SCHEMA DROP INDEX.
 - For Siebel Industry application (SIA) upgrades, select option 13: OLD SCHEMA DROP INDEX.This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(@GPDRPIX)*.

- 3 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(@GPDRPIX)`. This job runs DDL to drop old-schema non-unique indexes.

NOTE: After running this job, verify that it was successful by reviewing the sysout in SDSF or another job output facility and making sure the RC=0.

- 4 After submitting the job, enter `cancel` on the command line or press PF3 to save changes.
- 5 Verify that the JCLTEST return code is FLUSH (RC=0 or RC=4). If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

After the `###DRPIX` job has *successfully completed*, run the PRE-GEN PRIMARY INDEX DDL job to create new index schemas.

To create new index schemas

- 1 Make sure you are at the Siebel Upgrade Job Submission Menu.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 2 Select the option for the application you are upgrading.
 - For Siebel Business application upgrades, select option 11: PRE-GEN PRIMARY INDEX DDL.
 - For Siebel Industry application (SIA) upgrades, select option 14: PRE-GEN PRIMARY INDEX DDL.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(@GPPREIX)`.

- 3 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(@GPPREIX)`. This job runs DDL to create new-schema, non-unique indexes.

Verify that the JCLTEST return code is FLUSH (RC=0 or RC=4). If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I. Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 4 After submitting the job, enter `cancel` on the command line or press PF3 to save changes.

Rebuilding the Gen Primary 7.8 Indexes

Perform the following task to submit the rebuild jobs for the 7.8 Gen Primary indexes.

To rebuild a primary index

- 1 Make sure you are at the Siebel Upgrade Job Submission Menu for your upgrade path.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 2 Select the option for the application you are upgrading.
 - For Siebel Business application upgrades, select option 12: PRE-GEN PRIMARY INDEX REBUILDS.
 - For Siebel Industry application (SIA) upgrades, select option 15: PRE-GEN PRIMARY INDEX REBUILDS.

This places you in edit mode for dataset `DSNHLQ.SIEBEL.INSTALL.JCL(SUBGPRES)`.

- 3 Run the job using the JCL in dataset `DSNHLQ.SIEBEL.INSTALL.JCL(SUBGPRES)`.

Verify that the job ran successfully (RC=0 or RC=4). Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe”](#) on page 253.

- 4 After submitting the job, enter `cancel` on the command line or press PF3 to save changes.

This submits 2000+ Index Rebuild jobs. The number of Index Rebuild jobs varies by upgrade type. Job names range from `###10001` to `###1nnnn`.

Migrating the Gen Primary 7.8 SQL

Perform the following procedure to run the data migration scripts for Gen Primary.

To migrate the primary index SQL

- 1 Make sure you are at the SIEBEL UPGRADE JOB SUBMISSION MENU for your upgrade path.
 - For Siebel Business application upgrades, the panel ID is SBLRUNHP.
 - For Siebel Industry application (SIA) upgrades, the panel ID is SBLRUNFP.
- 2 Select the option for the application you are upgrading.
 - For Siebel Business application upgrades, select option 13: GEN PRIMARY.

- For Siebel Industry application (SIA) upgrades, select option 16: GEN PRIMARY.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBGENP)`.

- 3 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(SUBGENP)`.

After the job is run, verify that the job ran successfully.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs with an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 4 After submitting the job, enter cancel on the command line or press PF3 to save changes.

This submits 45 GenPrim job streams (the number of job streams varies by upgrade type). All 45 jobs can run in parallel—no dependencies exist.

RC=0, RC=4, or RC=8 indicates that the job was successful. RC=8 is associated with jobs with an X in position 3 of the member name.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 5 View the job status.

For information on how to view job status if you selected Siebel-scheduled mode, see [“Viewing the Siebel Job Log Status” on page 258](#).

- 6 After all GENPRIM jobs have completed successfully, run the POST-GEN PRIMARY INDEX DDL job.

Creating the New EIM Indexes

Perform the following procedure to create the new EIM indexes.

NOTE: This topic applies to Siebel Industry applications upgrades.

To create the new EIM indexes

- 1 Make sure you are at the Siebel Upgrade Job Submission Menu for your upgrade path.
The panel ID is SBLRUNFP.
- 2 Select the option for the application you are upgrading:

- For Siebel Business application upgrades, select option 14: EIM INDEX DDL.
- For Siebel Industry application (SIA) upgrades, select option 17: EIM INDEX DDL.

This places you in edit mode for dataset `DSNHLO.SIEBEL.INSTALL.JCL(@EIMX)`.

3 Run the job using the JCL in dataset `DSNHLO.SIEBEL.INSTALL.JCL(@EIMX)`.

4 Verify that the job ran successfully.

RC=0 indicates that the job was successful.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe”](#) on page 253.

5 After submitting the job, enter `cancel` on the command line or press PF3 to save changes.

6 Verify that the JCLTEST return code is FLUSH (RC=0 or RC=4). If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.

This job runs DDL to Create New-Schema (Siebel 7.8) EIM Indexes (Unique and Non-Unique).

Generating RUNSTATS

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

The following procedure generates RUNSTATS jobs for tablespaces that contain multiple-tables-per-tablespace or partitioned tablespaces and have *not* had statistics collected during any of the previous upgrade processes.

NOTE: This process excludes *all* interface tables (EIM and tables with an `_IF` suffix).

To generate RUNSTATS jobs

1 Make sure you are at the Siebel Upgrade Job Submission Menu for your upgrade path.

The Panel ID is SBLRUNFP.

2 Select the appropriate menu option for your upgrade path for intersection table maintenance:

- For Siebel Business application upgrades, select option 15: RUNSTATS (SCHEMA W/O STATS).
- For Siebel Industry application upgrades, selection option 18: RUNSTATS (SCHEMA W/O STATS).

3 Press Enter to start generating RUNSTATS jobs.

- 4 To start the generated RUNSTATS jobs, access the *DSNHLO.SIEBEL.INSTALL.JCL(SUBRUNST)* member and start the *###SUBRS* job.

This will start the jobs listed in the *DSNHLO.SIEBEL.EXEC(@RST)* member.

NOTE: If you are performing a development environment upgrade, run the RUNSTATS jobs before starting the repository merge process.

Performing Intersection Table Maintenance

Upgrades: All upgrades.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The intersection table maintenance jobs (ITM) identify and resolve potential duplicate row IDs from intersection tables.

To run the intersection table maintenance process

- 1 Make sure you are at the Siebel Upgrade Job Submission Menu for your upgrade path.
The Panel ID is SBLRUNFP.
- 2 Select the appropriate menu option for your upgrade path for intersection table maintenance:
 - For Siebel Business application upgrades, select option 16: INTERSECTION TABLE MAINT.
 - For Siebel Industry application upgrades, selection option 19: INTERSECTION TABLE MAINT.
- 3 On the ITM Maintenance Process Menu (Panel ID SBLITMP), select option 0: CHANGE JCL TEMPLATES - SET DSNHLO (REQUIRED), and then press Enter.
- 4 On the ITM Maintenance Process Menu (Panel ID SBLITMP), select option 1: BUILD/POPULATE PDS DATASETS (UNPACK03), and then press Enter.

This places you in edit mode for dataset *DSNHLO.SIEBEL.INSTALL.JCL(UNPACK03)*.

- 5 Run the job using the JCL in dataset *DSNHLO.SIEBEL.INSTALL.JCL(UNPACK03)*.

This job populates (unpacks) all ITM JCL and SQL PDS members into their corresponding PDS datasets.

DSNHLO.SIEBEL.ITM.SQL

DSNHLO.SIEBEL.ITM.JCL

- 6 Verify that the job ran successfully.

RC=0 indicates that the job was successful.

Verify that the JCLTEST return code is FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF2061.

Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe” on page 253](#).

- 7 After submitting the job, enter cancel on the command line or press PF3 to save changes.

- 8 On the ITM Maintenance Process Menu (the panel ID is SBLITMP), select option 2: ADD JOBCARDS TO INTERSECTION TABLE MAINTENANCE (ITM) JOBS, and then press Enter.

- 9 At the prompt, enter a three-character job name prefix, and then press Enter.

NOTE: The job name prefix is used for the ITM JCL generated for the Update and Delete jobs.

A message indicates that the JCL generation is in process, then a series of messages indicates the number of jobs generated. The number of jobs varies by upgrade path.

- 10 When three asterisks (***) appear at the bottom of the display messages, press Enter to return to the ITM Maintenance Process Menu.

The jobcards are added to the JCL templates.

- 11 On the ITM Maintenance Process Menu (the panel ID is SBLITMP), select option 4: RUN UNLOAD JOBS, and then press Enter.

This places you in edit mode for dataset `DSNHLQ.SIEBEL.INSTALL.JCL(SUBITMUL)`.

TIP: If you want to run the ITM Unload jobs manually, select option 3 to view the PDS list of all unload jobs. Option 4 submits all ITM Unload jobs in a single job submission.

- 12 To run the job using the JCL in dataset `DSNHLQ.SIEBEL.INSTALL.JCL(SUBITMUL)`, enter submit on the command line and then press Enter.

This job automatically submits all the unload jobs references in datasets `DSNHLQ.SIEBEL.EXEC(@ITMUL)`.

- 13** Verify that this job (and *all* jobs submitted automatically) ran successfully.

RC=0 or RC=4 indicates that the job was successful. Jobs with an ABEND (end abnormally) U99 or U099 message are jobs that failed.

Verify that the JCLTEST return code is FLUSH. If you do not see the FLUSH return code, you can verify the condition codes by searching for the condition code IEF206I.

If you selected Siebel-Scheduled Mode, this job is logged (as *Completed Successfully* or *Failed*) to the target Siebel log table: `TARGET_TABLEOWNER.TMP_SBLLOG_TAR`.

NOTE: You must fix any failed jobs before proceeding with the upgrade. For information on restarting failed jobs, see [“Restarting Failed Jobs on the Mainframe”](#) on page 253.

TIP: To review a list of all jobs submitted automatically, view datasets `DSNHLQ.SIEBEL.EXEC(@ITMUL)`. This corresponds to all PDS members listed in option 3.

- 14** After submitting the job, enter cancel on the command line or press PF3 to save changes.

- 15** On the ITM Maintenance Process Menu (the panel ID is SBLITMP), select option 5: RUN DELETE/UPDATE JOBS USING PDS LIST, and then press Enter.

This places you in edit mode for dataset `DSNHLQ.SIEBEL.ITMJOBS.JCL`. This is a PDS, so all PDS members are listed.

NOTE: If a message indicates that no maintenance jobs were generated, either option 4 was not run or no tables require maintenance.

- 16** To run each PDS member job using the JCL in dataset `DSNHLQ.SIEBEL.ITMJOBS.JCL`, select a PDS member, enter submit on the command line and press Enter. Repeat for each PDS member.

All jobs can run in parallel (that is, at the same time).

- 17** Verify that this job (and *all* jobs submitted automatically) ran successfully.

- 18** After submitting the job, enter cancel on the command line or press PF3 to save changes.

Resuming the Siebel Upgrade Wizard on the Midtier After the Third Pause

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Once you have completed the tasks outlined in [“Executing the Upgrade on the z/OS Host \(Phase 2 of 2\)”](#) on page 243 and have completed the task in [“Performing Intersection Table Maintenance”](#) on page 249 (Production environment), continue the upgrade from the point at which it paused.

To resume the Siebel Upgrade Wizard on the midtier after the third pause

- 1** Resume the upgrade using the appropriate method for your operating system.

Windows:

- If the dialog box is still open from [Step 2 on page 239](#), click Yes in the dialog box for Pause #3.
- If you closed the Siebel Upgrade Wizard screen or it is no longer open, restart the upgrade by typing the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\BIN` directory:

```
si ebupg.exe /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
si ebupg.exe /m master_upgdev_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
si ebupg.exe /m master_upgdev_prod_704_mf.ucf
```

UNIX:

- Navigate to `$SIEBEL_ROOT/bin` and enter the following command:

```
srvrupgz /m master_UPGRADEOPTION_ENVIRONMENT_VERSION_mf.ucf
```

For example, to restart a development environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgz /m master_upgdev_dev_704_mf.ucf
```

To restart a production environment upgrade from Release 7.0.4, enter the following command:

```
srvrupgz /m master_upgdev_prod_704_mf.ucf
```

- 2 In the Siebel Upgrade Wizard screen, select Yes to resume the upgrade.

NOTE: When you are performing a development environment upgrade, the Siebel Upgrade Wizard may stop responding during the Repository Data Upgrade step. If this happens, cancel the Siebel Upgrade Wizard and, from the command prompt, navigate to \$Si ebsrvr\B i n and execute the RUNSTATS commands on tables S_COLUMN and S_UK_ATTJOIN using the following syntax:

```
RSTAT390 /u <database_username> /p <database_password> /c <ODBC_datasource> /d
<Tableowner> /a Y /l <logfile> /T S_COLUMN
```

```
RSTAT390 /u <database_username> /p <database_password> /c <ODBC_datasource> /d
<Tableowner> /a Y /l <logfile> /T S_UK_ATTJOIN
```

After executing the these commands, restart the Siebel Upgrade Wizard from the command prompt by entering siebug.exe /m master_upgrep_dev_<version>_mf.ucf.

The remaining upgrade jobs run on the midtier. A check mark appears beside each item on the Upgrade Wizard screen as it is completed.

CAUTION: The upgrade process executes in a DOS command window. Do not click inside this window while the upgrade is running. If you accidentally click inside the DOS command window, the process pauses and the word *Select* precedes the title of the command window. To continue, press the space bar or Escape.

- 3 Select OK.

NOTE: (Development upgrades only.) At this point, the mainframe-centric upgrade defaults to the midtier-centric approach to the upgrade. That is, differences between the two types of development upgrade exist only for the upgrep phase and only for the steps before this step. When you restart the upgrep process on the midtier after the third pause, the same sequence of steps are run for both mainframe-centric and midtier-centric development upgrades.

Restarting Failed Jobs on the Mainframe

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This topic describes how to restart jobs that fail when you are performing a mainframe-centric upgrade.

To restart a failed job on the mainframe

- 1 Identify the job that failed under the SDSF exit (job status).

If you are using Siebel-scheduling, you can find the name of the job that failed using option 6: JOB LOG - DISPLAY JOB STATUS (SIEBEL SCHEDULED MODE) on the SBLUPG78 panel. See [“Viewing the Siebel Job Log Status” on page 258](#) for further information.

2 Determine the reason the job failed. You can determine the reason for the job failure by selecting the job on the SDSF output queue panel using the s action character.

3 Correct the problem.

If a job fails because of an SQL error, fix the problem in the appropriate SQL PDS member. For example, for PRESCHM jobs, you can locate the relevant SQL PDS member in the DSNHLO.SIEBEL.PRESCHM.SQL data set. You can also find information about which member corresponds to a failed job using option 6: JOB LOG - DISPLAY JOB STATUS (SIEBEL SCHEDULED MODE) on the SBLUPG78 panel.

NOTE: Contact Siebel Technical Support if you need help in performing these tasks or if you require confirmation that the tasks that you are about to run are correct.

4 Once the problem has been identified and corrected, restart the job. You can do this by selecting the job that failed on the SDSF output queue using the SJ action character. This will automatically call up the next scheduled job allowing the upgrade process, for example PRESCHM, to continue.

14 Reviewing the Database Upgrade Log Files

This chapter describes how you can use the log files that the Siebel Upgrade Wizard produces during the upgrade process. It includes the following topics:

- [“About the Database Upgrade Logs”](#)
- [“Reviewing Upgrade Log Files for Errors” on page 256](#)
- [“Reviewing Log Files for Errors in Jobs Run from the Midtier” on page 258](#)
- [“Viewing the Siebel Job Log Status” on page 258](#)
- [“Running SQL in Siebel Logs” on page 259](#)
- [“Manually Archiving Upgrade Log Files” on page 259](#)

About the Database Upgrade Logs

Upgrades: All upgrades.

Environments: All environments.

The Siebel Upgrade Wizard writes logs that provide detailed information on the upgrade processes and any errors that occurred when the processes were running. The Upgrade Wizard writes the logs for a process to the following directory by default:

Windows: `SIEBEL_ROOT\log\PROCESS`

UNIX: `$SIEBEL_ROOT/log/PROCESS`

where *PROCESS* is the name of the upgrade process you have run, for example, `upgrep_dev_704` or `prepare_for_product_i on_upgrade`.

NOTE: You can select a different log directory from the Log Output Directory screen on the Database Server Configuration utility.

The *PROCESS* directory contains the following subdirectories:

- **Output.** Directory containing the Upgrade Wizard log files.
- **State.** Directory containing the state.log file.

About the State Log File

Each upgrade process consists of a series of steps, each of which must complete successfully. If the Upgrade Wizard cannot complete a step, it marks the step as incomplete in the state.log file and exits. The state.log file is located in the `SIEBEL_ROOT\LOG\process\state` (Windows) or `$SIEBEL_ROOT/LOG/process/state` (UNIX) directory.

You must correct the error and then 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.

About Process Log Files

You can identify errors you encounter during an upgrade by reviewing the log file named UpgWiz.log (Windows) or srvrupgwiz1.log (UNIX) in the `SIEBEL_ROOT\LOG\process\output` directory (Windows) or the `$SIEBEL_ROOT/LOG/process/output` directory (UNIX).

The name of the log file increments for subsequent log files that are created if the Siebel Upgrade Wizard encounters a problem and you have to run it again.

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

How to Determine if the Upgrade Process Completed Successfully

If the status of all the steps in the `state.log` is Complete, the upgrade process completed successfully.

If the status of any step is Incomplete, the upgrade process did not complete successfully. You must identify the error and correct it before resuming the upgrade.

NOTE: 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, even those with a status of Complete.

Use the following process to identify errors:

- 1 Resolve errors for steps identified with a Status of Incomplete in the `state.log` file.
- 2 Review all the steps with a status of Complete in the `state.log` file. If any contain unacceptable errors, resolve these errors. See [“Reviewing Upgrade Log Files for Errors”](#) for information on identifying unacceptable upgrade errors.
- 3 Restart the Upgrade Wizard, or, if necessary, restore the database and rerun the upgrade process.

If you have any questions about how to resolve errors, contact Siebel Technical Support.

Log Files That Can Be Ignored

If the upgrade completed successfully, there are several log files that you can safely ignore:

- Windows: `sw_cfg_xxx.log` and `siebel.log`
- UNIX: `srvrupgwiz_*.log` and `siebel_*.log`. For example, `srvrupgwiz_001.log`, and `srvrupgwiz1_02.log`
- Any other log file that existed before the start of the upgrade

CAUTION: UNIX Only: The log file `srvrupgwiz_001.log` is a different file to the `srvrupgwiz1.log` file. Do not ignore log files named `srvrupgwiz1.log`, `srvrupgwiz1_01.log`, and so on.

Reviewing Upgrade Log Files for Errors

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

You should review the logs created when you run the Siebel Upgrade Wizard to verify that the upgrade process completed correctly and to identify errors that must be resolved. The log files may include errors that are expected and benign. You must compare any error messages found in the log files to a list of acceptable error messages, and correct any non-benign errors.

Complete the following procedure to manually review log files for unacceptable errors.

Reviewing the Log Files

Complete the following procedure to manually review log files for unacceptable errors.

To manually review the log files for unacceptable errors

- 1 Review the state1.log file to see at which step the upgrade failed. This step can be traced back to the driver file. The state1.log file is located in the following directory:

Windows: `SI EBEL_ROOT\log\PROCESS\state`

UNIX: `$SI EBEL_ROOT/ log/ PROCESS/state`

- 2 Print the errors file. The errors file lists the benign and expected errors you might find in the log files; you can ignore these errors. The errors file is located in the installation subdirectory:

Windows: `DBSRVR_ROOT\DB2390\errors.rtf` or `errors.htm`

UNIX: `DBSRVR_ROOT/DB2390/errors.txt`

- 3 Sort the files in the following directory by date.

Windows: `SI EBEL_ROOT\log\PROCESS\output`

UNIX: `$SI EBEL_ROOT/ log/ PROCESS/output`

- 4 Open each log file, starting with the earliest, and search for errors. Starting with the earliest log file can shorten your research time.

Log files are identified by the .log extension. Errors are either tagged with the word *error* or enclosed in square brackets [...].

- 5 For each error found, compare the error description against the list of acceptable errors documented in the errors file.

The log files generated by the Siebel Upgrade Wizard (for example `svrupgwiz1.log`) appear in the errors file as `upgwiz1.log`, `upgwiz2.log`, incrementing for additional log files. Identify errors as follows:

- If you find the error in the errors file, it is acceptable and no action is required. Continue to review the errors found in the log file.

- If an error appears multiple times in a log file, but only one occurrence of that error appears in the errors file, all errors of that type are acceptable and no action is required. Continue to review the errors found in the log file.
- If a log file is not listed in the errors file, there are no acceptable error messages for that log file. You must correct the condition that caused the error before you rerun the Siebel Upgrade Wizard.
- If you find an error that is not listed in the errors file, it is unacceptable. You must correct the condition that caused the error before you rerun the Siebel Upgrade Wizard.

To help resolve unacceptable errors, view 477324.1 (Article ID) on My Oracle Support. This document was formerly published as Siebel 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, create a service request (SR) on My Oracle Support. Do not proceed with the upgrade.

6 Repeat [Step 5 on page 257](#) for each log file.

Although non-benign errors are rarely encountered, this review is critical. Certain errors, such as a failure to create indexes, may result in performance problems or anomalous behavior in Siebel Business applications.

Reviewing Log Files for Errors in Jobs Run from the Midtier

Upgrades: All upgrades.

Environments: Development (midtier-centric) environment.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

To find errors in the log files for jobs run from the midtier during an upgrade, search on the word SQLSTATE.

Viewing the Siebel Job Log Status

Upgrades: All upgrades.

Environments: Production test, production.

If you are using Siebel-scheduling to run your upgrade jobs, you can query the Siebel job log by completing the following procedure.

NOTE: To view job status, you must have installed DSN REXX.

To view the Siebel job log status

1 If you are not on the Siebel Upgrade Main Menu, enter the following command:

```
TSO SBLUPG78
```

- 2 Select option 6: JOB LOG - DISPLAY JOB STATUS (SIEBEL SCHEDULED MODE).
The SIEBEL JOB LOG QUERY panel displays. The panel ID is SBLLOGP.
A list of successful and failed jobs appears.
- 3 Next to the label LIST BY JOB TYPE/STATUS: , enter 2 or 9 for FAILED jobs. Enter option 1 to list the UNLOADS jobs.
The list displays 250 lines only. The Unload, Load and Index Rebuild processes have more than 250 jobs, so you must query using another option or by specific or partial job name.
- 4 Press PF3 when you are finished viewing the log.
The Siebel Upgrade Main Menu for your upgrade path appears.

Running SQL in Siebel Logs

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

By using SPUFI or the command line, you can construct SQL queries to run against the source or the target log tables.

The following statements report the status of the load jobs for the source or target databases:

- ```
SELECT JOB_DESC, JOB_NAME, JOB_STATUS FROM CQ10A901.TMP_SBLLOG_SRC WHERE JOB_NAME LIKE ' LKC%';
```
- ```
SELECT JOB_DESC, JOB_NAME, JOB_STATUS FROM CQ10A901.TMP_SBLLOG_TAR WHERE JOB_NAME LIKE ' LKC%';
```

You can alter the preceding statements to report the status of any jobs by changing the LIKE JOB_NAME statement to another prefix.

This following statement checks for failed unload jobs, but can check for any other job by changing the JOB_NAME LIKE statement to use the appropriate prefix.

```
SELECT JOB_DESC, JOB_NAME, JOB_STATUS FROM CQ10K034.TMP_SBLLOG_SRC WHERE JOB_STATUS != ' COMPLETED SUCCESSFUL' AND JOB_NAME LIKE ' LKB%';
```

Manually Archiving Upgrade Log Files

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After a successful installation and upgrade, you must manually save and archive the log files located in the *SIEBEL_ROOT/1og/PROCESS* (Windows) directory.

By default, only nine (9) upgrade log files are retained for subsequent retries of the Siebel Upgrade wizard. After nine log files have been created, when the Siebel Upgrade Wizard is rerun, it overwrites the log files, beginning with the earliest file 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 `siebel_log_archive` environment variable. For example, set the `siebel_log_archive` environment variable to 20 to retain twenty log files.

15 Performing the Siebel Tools Repository Merge

This chapter lists the steps involved in preparing for and performing a repository merge during a development environment upgrade. It includes the following topics:

- [“About Inheriting Upgrade Behavior”](#)
- [“About the Postmerge Utilities” on page 264](#)
- [“About the Incorporate Custom Layout \(ICL\) Upgrade Option” on page 264](#)
- [“Configuring Objects to Inherit Upgrade Behavior” on page 270](#)
- [“Migrating Repository Objects to the Standard UI” on page 271](#)
- [“About Backing Up the New Customer Repository or Database Schema” on page 275](#)
- [“About Reorganizing Tables Before the Repository Merge” on page 275](#)
- [“Running the Repository Preparation Wizard on Release 6.2.1 Repository” on page 277](#)
- [“Performing a Repository Merge” on page 279](#)
- [“Determining If a Repository Merge Was Successful” on page 288](#)
- [“Reviewing Attribute Conflicts in the Repository Merge” on page 291](#)
- [“Reviewing Deleted Objects in the Repository Merge” on page 292](#)
- [“Reviewing Obsolete Objects in the Repository Merge” on page 293](#)
- [“Generating EIM Temporary Columns After a Repository Merge” on page 294](#)
- [“Regenerating the Repository Definition Files” on page 295](#)
- [“Deleting Unneeded Repository Files” on page 297](#)
- [“Creating a New SRF File” on page 297](#)

About Inheriting Upgrade Behavior

Upgrades: All upgrades.

Environments: Development environment only.

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 Siebel Systems. Objects in the Prior Siebel Repository and the New Siebel Repository are standard objects.

Typically, you create customized objects by making a copy of a standard object and then modifying the copy, called the *descendent*.

You can specify upgrade ancestors for the following object types:

- Applets
- Business Components
- Integration Objects
- Reports

You specify the upgrade ancestor in the descendent's Upgrade Ancestor field in Siebel Tools. During the repository merge, the descendent is upgraded in the same way as the upgrade ancestor.

For example, you copy the standard object Applet A and name the descendent Applet B. In the New Siebel Repository, Applet A has been modified to include several new list columns.

Here is what happens to Applet B (the descendent) during the repository merge:

- If you specified Applet A as the upgrade ancestor, then Applet B is upgraded to include the two new list columns.
- If you did not specify Applet A as the upgrade ancestor, then Applet B does not receive the two new list columns.

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 not present in the New Siebel Repository, it is obsolete, and its upgrade behavior is not propagated to descendents. If an ancestor object is obsolete in the New Siebel Repository, descendents are not also obsolete. They are copied to the New Customer Repository.
- 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, descendents are not modified.
- The Upgrade Ancestor Property is not used by the postmerge utilities. The postmerge utilities revise layout objects without regard to how ancestor objects were revised.
- 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 descendent during the repository merge.

You can manually propagate changes to descendents 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 Topic

["About the Postmerge Utilities" on page 264](#)

About the Postmerge Utilities

Upgrades: All upgrades.

Environments: Development environment only.

You run the postmerge utilities after the repository merge completes. The postmerge utilities make revisions to layout objects from the Prior Customer Repository that have been added to the New Customer Repository:

- **User interface navigation.** Release 7.7 introduced a revised method for UI navigation. The postmerge utilities make revisions to new or customized layout objects from the Prior Customer Repository to integrate them into the new UI navigation scheme.
- **Multi-Value group applets.** The postmerge utilities enable the shuttle feature for multi-value group (MVG) applets. MVG applets must meet the criteria described in *Configuring Siebel Business Applications* for MVG shuttle applets.
- **Flow-Based applets.** The postmerge utilities convert many flow-based form applets to grid-based applets. If you have customized a form applet by adding fields and controls, the postmerge utilities place these at the bottom of the applet. You must reposition them after the merge.

How the Postmerge Utilities Work with Objects with Upgrade Ancestors

The postmerge utilities do not use the Upgrade Behavior property. This includes the following changes:

- Enabling the shuttle feature for MVG applets
- Converting applets to grid-based layout.

The postmerge utilities make these changes to applets based on their characteristics rather than upgrade behavior. Some customized applets may not be converted. See *Configuring Siebel Business Applications* for a full description of the characteristics applets must have before these changes can be applied. *Configuring Siebel Business Applications* also describes how to manually convert applets to grid-based layout.

Related Topic

[“About Inheriting Upgrade Behavior” on page 261](#)

About the Incorporate Custom Layout (ICL) Upgrade Option

Upgrades: Release 7.x using ICL.

Environments: Development environment only.

If you are upgrading from Siebel 7.x, you can choose to preserve the layout of applets and views. This reduces the workload required to reconfigure customized UI objects after upgrading to a new release that has significant UI changes.

For example, if you are upgrading from Release 7.0.x or Release 7.5.x, you can preserve Release 7.0.x or Release 7.5.x layouts when you upgrade.

An ICL upgrade preserves the layouts of both standard and customized UI objects.

When to Use ICL

To determine when to use ICL, assess both your current implementation and your future implementation plans.

Use the following factors to assess your current implementation:

- The higher the number of customizations of standard UI objects, the greater the benefit from choosing ICL.
- If you have customized only a few standard UI objects, choosing ICL may not be cost effective.
- If you have customized many standard UI objects but these objects are in applications that do not support ICL, performing an ICL upgrade may not be cost effective.

It is important to assess your future development plans because you can select ICL for only every other upgrade. This means you should choose ICL for upgrades where your customizations are heaviest.

For example, your current release has a moderate number of customized standard UI objects. At the next release, you plan to make heavy customizations of standard UI objects. Consider not choosing ICL when upgrading your current release. This enables you to choose ICL at the next release, when your customizations will be heavier.

You Can Preserve Layouts for Only One Upgrade

You can preserve a release's layouts for only one upgrade.

If you select ICL for an upgrade, then at the following upgrade, the repository merge process requires that you do the following:

- Before the repository merge, you must upgrade the UI objects preserved in the prior ICL upgrade to the standard objects for the release from which you are upgrading. This brings your UI forward to the installed release.
- Perform the repository merge for the new release. Selecting ICL for this merge is not recommended.
- After the upgrade to the new release, you must reconfigure customized layouts for applets, views, and screens.

How Layouts Are Preserved

When you select ICL during a repository merge:

- The layout of applets in the Prior Customer Repository is preserved.
- The location of controls in applets is preserved.

- Except as noted later in this topic, no new controls from the New Siebel Repository are added to applet layouts.
- The association of applets to views is preserved.
- The layout of views is preserved.
- The association views to screens is preserved.
- View locations and groupings within screens are similar but not identical.
- Changes to navigation in the release you are upgrading to are implemented.

For example, in Release 7.7, a new navigation scheme was introduced. If you are upgrading from a release prior to 7.7, this navigation scheme is implemented during upgrade regardless of whether you choose ICL. If controls in applets conflict with the new navigation scheme, they are handled as described later in this topic.

How the Repository Manages an ICL Upgrade

Repository objects are affected as follows during an ICL merge:

- View and applet Web templates, including some child objects, in the New Customer Repository are inactivated and made read-only. Their names are appended with the string -UPG. You cannot delete these objects from the repository or change their modification time.
- The inactivated Web templates and child objects are replaced with corresponding records from the Prior Customer Repository. This preserves the layout of the release from which you are upgrading.

The ICL Option Does Not Affect All Repository Objects

Since Release 7.7, Siebel Tools objects have a property called Upgrade Behavior:

- If this property is set to NULL or Preserved and ICL is selected during repository merge, ICL preserves the layout of the previous release.
- If this property is set to ADMIN, then ICL does not attempt to preserve the layout of the previous release. Layouts in the current release are applied to the object. The Upgrade Behavior property is set to read-only when the value is ADMIN.

The following types of objects have the Upgrade Behavior property set to ADMIN. The layout of these objects is not preserved even if ICL is selected during the repository merge:

- Siebel administrative views and applets where core functionality must be upgraded to include feature enhancements. In some cases, these enhancements have such a large impact on the user interface that preserving layouts from a prior release is not practical.
- Screens and views in Siebel products that are significantly enhanced in a release:
 - **Release 7.7:** Applets and views for Siebel Employee Relationship Management (ERM) and Siebel Marketing.
 - **Release 7.8:** Applets and views in the Quotes and Orders screens and related customer management applets and views.

How ICL Affects Specific Repository UI Objects

When you select the ICL option, you use a new set of Web templates and certain child objects that preserve the layout of the release from which you are upgrading. ICL affects UI objects in the ways described in the following sections. This applies only to UI objects that do not have Upgrade Behavior set to ADMIN.

General UI

The following characteristics apply to uncustomized and customized UI objects in the Prior Customer Repository when you choose an ICL upgrade:

- Layouts are preserved.
- Attributes defined on Web templates are preserved. This includes relative positioning of fields and layouts.
- Properties of controls or list columns are not preserved. For example, height, width, caption, and pop-up icons are not preserved.
- ICL preserves existing controls. No controls are removed. Location of controls is also preserved.
- System pages such as Help, About View, About Record, and Technical Support are not preserved.
- Font style, color, text size, and object alignment are not preserved.
- Stylesheets are not preserved. Web template files are approximations of their 7.x counterparts.

Applets

The following characteristics apply to uncustomized and customized applets in the Prior Customer Repository when you choose an ICL upgrade:

- Associations to views are preserved.
- The upgrade converts certain form-based applets to grid layout. If ICL is selected, the postmerge utilities converts them back to flow-based.
- Applet drilldown child objects are preserved.
- Applet-level and application-level menus are not preserved.
- For flow-based applets, field labels are placed on top as in 7.0.x and 7.5.x However, you can choose to put the labels on the left when setting up the merge.
- In Release 7.8, the New File and New URL buttons are added to attachment applets. If you select ICL, these buttons are not added to applets.
- In Release 7.8, MVG dialog boxes display an OK button to close them. If you select ICL, MVG dialog applets have both OK and Cancel buttons.
- New buttons from the New Siebel Repository are not added to applets. You must add these buttons as desired after completing the upgrade.
- In Release 7.8, the UI contains the Query Assistant and Quick Print features. If you select ICL, these buttons are not added to applets. You must add these buttons as desired after completing the upgrade.

- Release 7.8 applets have three default buttons—New, Delete, and Query. If you select ICL, applets keep their existing default buttons.
- In Release 7.8, the Reset button is not displayed on applets. If you select ICL, applets will keep the Reset button.
- In Release 7.8, Pick and Association applets support in-line queries. This feature is implemented regardless of whether or not you choose ICL.

Views

The following characteristics apply to uncustomized and customized applets in the Prior Customer Repository when you choose an ICL upgrade:

- Applets appear in the views they reference.
- Their location in the view is preserved.
- View Web template child objects are preserved.
- On parent list views, view tabs do not display in a row near the middle of the view as they do in 7.0.x and 7.5.x. You must select a record in the parent list applet to display the view tabs.

Screens

The following characteristics apply to uncustomized and customized screens in the Prior Customer Repository when you choose an ICL upgrade:

- Screens contain the views they reference.
- The view locations and groupings will be similar but not identical.

Navigation

The navigation method used in Siebel 7.8 is applied to the UI regardless of whether or not you select ICL during the repository merge.

How ICL Affects Web Template Files and the Style Sheet File

When you upgrade to the new release, each Siebel Server and Siebel Tools installation receives a fresh set of Web template files. Each installation also receives a new style sheet file (`main.css`). These files help define the look and feel of the new release.

They are located as follows in the Siebel Tools installation directory (Windows path syntax):

- Web template files directory: `\webtempl`
- Style Sheet file: `\publi c\lang\files\main.css`

In the paths, *lang* is the installation language, for example `enu`.

When you select ICL during a merge, the postmerge utilities perform the following steps in the installation directory of the Siebel Tools that you used to perform the merge:

- Moves the new Web template files from `webtempl` to `\temp\webtempl`.

- Copies Web template files from `reppatch\web_templates` to `\webtempl`.

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 `reppatch`. These files provide the look and feel for Release 7.5.3.

These are the out-of-the-box Web template files for Release 7.5.3. You must evaluate customizations you have made to Web template files and determine whether to reimplement them in the Web template files in the new release.

When you select ICL, the postmerge utilities make no changes to the style sheet file installed with the new release:

- If you have customized this file, you must evaluate the customizations and determine whether to reimplement them in the style sheet file in the new release.

If you select the ICL option *Label On Top*, you must edit the style sheet file to obtain proper text-label alignment. This step is included in the process steps in [Chapter 4, “How to Perform the Upgrade.”](#)

How ICL Works with the Upgrade Ancestor Property

Use the Upgrade Ancestor property and ICL together as follows:

- For ICL and non-ICL upgrades, set the Upgrade Ancestor property on business component integration object and reports descendants. This ensures that your customizations to descendants are upgraded in the same fashion as the associated standard-object.
- If you are using ICL, set the Upgrade Ancestor property on descendant applets and child objects. This makes sure that ICL preserves the look and feel of your customizations.
- If you are not using ICL, setting the Upgrade Ancestor property on applets and child objects is not recommended. If the merge causes UI layout problems with the UI standard object, this prevents the problem from being propagated to the descendant UI object.

The postmerge utilities implement the ICL changes to objects in the New Customer Repository. This includes overwriting customizations to Web templates and Web template items. However, the postmerge utilities do not overwrite customizations to applets and control objects.

What Happens at the Next Release

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 active Web templates and child objects in the repository. (These are the objects that were preserved from the Prior Customer Repository.)
- Activates the Web templates and child objects with the `-UPG` suffix and removes the suffix. These `-UPG` objects are from the New Customer Repository for the current release (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 Web templates and child objects that have been customized since the prior upgrade. If an object has been customized since the prior upgrade, it is not replaced by its corresponding `-UPG` object. You must manually reconfigure these customized objects after the merge.

The postmerge utilities make no changes to the Web template files or style sheet file included in the new release.

Repository Merges That Do Not Use ICL

For upgrades from Release 7.x that do not use ICL, new Siebel objects and controls, usability enhancements, and layout changes are applied. This modifies the layout of applets, views, and screens and attempts to merge rather than preserve customizations.

After the upgrade, you will need to identify those areas of the upgraded interface where you want to reapply customizations.

If you have customized Web template files or the style sheet (main.css), these customizations are not merged into the Web template files or style sheet in the new release. You must evaluate these customizations and determine whether to reimplement them.

Related Topics

[“About Inheriting Upgrade Behavior” on page 261](#)

[“About the Postmerge Utilities” on page 264](#)

Configuring Objects to Inherit Upgrade Behavior

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

In Release 7.8, 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 descendents or copies of an object

- 1 Right-click on the object.
- 2 Select View Descendents from the picklist.
A dialog box appears and lists the descendents.

Migrating Repository Objects to the Standard UI

Upgrades: All upgrades. Perform this task if you selected Incorporate Custom Layout (ICL) on your previous upgrade.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If you selected the Incorporate Custom Layout (ICL) feature for your previous upgrade, you must migrate UI repository objects to the standard UI before doing the repository merge for your next upgrade. The migration does not affect customized objects in the Prior Customer Repository. Customized objects are preserved and are included in the repository merge.

Migrating to the standard UI uses defined logic to select objects and then modifies these objects as follows:

- **ICL UI objects:** Inactivates and deletes the Web templates and certain child objects. These are the objects that preserved the layouts of the previous release.
- **Standard UI objects.** Activates the UI Web templates and child objects that provide the standard UI look and feel for the release you are upgrading from. These objects have “-UPG” appended to the name.

For a full description of how ICL works during an upgrade, see [“About the Incorporate Custom Layout \(ICL\) Upgrade Option” on page 264.](#)

How Customizations Are Handled

ICL UI objects you have customized after the previous upgrade are not affected by the migration to the standard UI. These changes are preserved and are included in the repository merge for the new release.

Siebel Tools identifies these customizations by comparing the modification time of ICL UI objects with their corresponding standard UI objects. If the modification times differ by more than a specified lag time, Siebel Tools does not change the ICL UI object, and it is treated as a customized object in the upcoming repository merge.

For example, your previous ICL repository merge required about three days to complete. This means the modification time of an ICL UI Web template and its corresponding standard UI Web template did not differ by more than three days when the repository merge completed. You later modified the ICL UI Web template. Its modification time now differs by more than three days from the corresponding standard UI Web Template.

You then use Siebel Tools to migrate the repository to the standard UI and use a lag time of three days. Since the modification time comparison for the ICL UI Web template is greater than the lag time, the ICL UI Web template is not replaced by the corresponding standard UI Web template. The ICL UI Web template is treated as a customized object in the upcoming repository merge.

The default lag time is 72 hours. You can specify a lag time between 24 and 120 hours. Observe the following guidelines:

- Avoid setting a lag time that is shorter than the time required to complete the previous repository merge. This can cause objects that were not customized after the merge to be treated as customized.
- Do not set a lag time that is significantly larger than the length of the previous repository merge. This increases the risk that customizations were made before the lag time expired. These customizations will be lost during the upcoming repository merge.

How Repository Objects Are Changed

The migration process locates both standard UI objects and ICL UI objects in the repository. The process then determines whether to modify the standard UI object or the corresponding ICL UI object:

- Deleting the ICL UI object and activating the corresponding standard UI object migrates the UI to standard.
- Deleting a standard UI object and retaining the corresponding ICL UI object preserves the customized ICL UI object.

[Table 26 on page 273](#) shows the logic used to modify these objects. Interpret the table columns as follows:

- **ICL UI Object Found?** "Yes" means Siebel Tools has located a standard UI object, and it has a corresponding ICL UI object. No means that a corresponding ICL UI object is not in the repository. In most cases, this is because the object was inactive in the Prior Customer Repository during the previous merge. When this occurs, the object is appended with "-UPG." No ICL UI object that preserves look and feel is created.
- **Within Lag Time?** "Yes" means that the modification time comparison is within the specified lag time. No means the comparison is not within the specified lag time. No indicates that you have modified the ICL UI object after the repository merge.
- **Status in Prior Standard Repository.** Siebel Tools checks the Active/Inactive status in the Prior Standard Repository to prevent activating an object that is inactive in the Prior Standard Repository.

In the table, Prior Customer Repository refers to your current Siebel Repository, which you renamed in preparation for the upcoming repository merge.

Table 26. Logic Used to Modify Repository Objects

ICL UI Object Found?	Within Lag Time?	Status in Prior Standard Repository	How Migration Modifies Objects in the Prior Customer Repository
Yes	Yes	Active	<ul style="list-style-type: none"> ■ Deletes ICL UI object. ■ Removes -UPG suffix from name of corresponding standard UI object. ■ Changes object status from Inactive to Active in the Prior Customer Repository. ■ Removes read-only restriction. Object can be modified or deleted.
Yes	Yes	Inactive	<ul style="list-style-type: none"> ■ Deletes ICL UI object. ■ Removes -UPG suffix from name from corresponding standard UI object. ■ Status is not changed from Inactive to Active in the Prior Customer Repository. ■ Removes read-only restriction. Object can be modified or deleted.
Yes	No	N/A	<ul style="list-style-type: none"> ■ Makes no changes to ICL UI object. (Object has been customized after the repository merge.) ■ Deletes corresponding standard UI object and children of this object.
No	N/A	N/A	<ul style="list-style-type: none"> ■ Status is not changed from Inactive to Active. ■ Removes read-only restriction. Object can be modified or deleted.

How Logging Is Done

The migration process logs the changes to the repository in the following log file:

SIEBEL_ROOT\log\icl migration. log

The beginning of the log lists top-level objects that were affected by the ICL feature. For each object, the log then iteratively lists the operations performed on all child objects.

Specifying a Lag Time

The default lag time is 72 hours. You can revise this by editing the Siebel Tools .cfg file. The minimum is 24 hours, and the maximum is 120 hours.

Prerequisite: Your Siebel Tools installation must be release 7.7 or later.

To specify a lag time

- 1 In the Siebel Tools installation directory, navigate to the `\bin\lang` directory, where *lang* is the installed language, for example ENU.
- 2 Using a text editor, open `tool.cfg`, and locate the [Siebel] section.
- 3 Add the following variable at the end of the section:

```
PriorICLMergeTimeLag = time
```

where *time* is an integer greater than or equal to 24 and less than or equal to 120.
- 4 Save the file.
- 5 Restart Siebel Tools.

Migrating to the Standard UI

A menu option in Siebel Tools migrates the repository Web templates and child objects to the standard UI. Upgrading the UI requires a minimum of three hours to complete.

Prerequisites:

- Your Siebel Tools installation must be release 7.8.
- Set the lag time as desired. The default is 72 hours.
- Verify that you have backed up the upgraded database.

To migrate to the standard UI

- 1 Start Siebel Tools.
- 2 In the Tools menu, choose Upgrade. Then select Migrate ICL Objects to Standard.
- 3 In the ICL Migration dialog, make the following selections and click Continue:
 - **Prior Customer Repository:** Select Prior Customer Repository. This is the Siebel Repository in your current release, which you have renamed in preparation for the upgrade.
 - **Prior Standard Repository:** Select Prior V7.X Siebel Repository. This is the Prior Standard Repository for your currently installed release. This repository was loaded when you performed the database upgrade for the new release.

- 4 In the ICL Migration Warning dialog, click Yes to confirm you have backed up the Prior Customer Repository.

The ICL Migration Status dialog box appears. It displays log entries during the migration.

During the migration, you cannot perform other operations in Siebel Tools. You also cannot close Siebel Tools.

A pop-up message displays when the migration is complete.

- 5 To cancel the migration after it has begun, do the following:
 - a In the ICL Migration Status dialog box, click Cancel.
 - b In the dialog box that asks you to confirm you want to cancel, click Yes.

If you cancel the migration after it has begun, you must restore the saved Prior Customer Repository from backup and begin the migration again.

About Backing Up the New Customer Repository or Database Schema

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The process of merging repositories to create the final customized repository used in the upgrade is time-intensive and resource-intensive. As a result, a merge might sometimes fail because of environmental factors, for example, space constraints. When this happens, the merge process continues, even if there is a fatal database error, and the errors might not be detected for some time.

If the merge fails, you must restore the database environment to its pre-merge state and run the merge again. There are two methods you can use to preserve the pre-merge environment so that you can restart the merge again if you need to. Before beginning the merge you can either:

- **Backup the entire database schema** then, if the merge fails, you can then restore the schema and rerun the merge. The disadvantage of this option is that it is time consuming and disk space intensive.
- **Export the New Customer Repository** to create a backup copy. This is a simpler option. If the merge fails, simply import the New Customer Repository. See *Using Siebel Tools* for information on exporting and importing repositories using the Database Server Configuration utility.

About Reorganizing Tables Before the Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

During the repository merge process, objects from four separate repositories are read and compared. Because this is a memory-intensive process, it is recommended that you execute the REORG utility on certain tables before performing the repository merge to improve performance. The following tables receive a large number of inserts during each repository import; running REORGs on each table's ROW_ID column will significantly increase the performance of the merge.

- S_APPLET_METH_MI
- S_APPL_WEB_TMPL
- S_APPL_WTMPL_IT
- S_BOCOMP
- S_BUSCOMP_UPROP
- S_COLUMN
- S_CONTROL_INTL
- S_CONTROL_UPROP
- S_DDOWN_OBJECT
- S_EIM_FK_MAPCOL
- S_FIELD
- S_INDEX
- S_INDEX_COLUMN
- S_INTFLD_UPROP
- S_INT_CKEY_FLD
- S_INT_COMP
- S_INT_FIELD
- S_JOIN
- S_JOIN_SPEC
- S_LIST
- S_LIST_COLUMN
- S_PICKMAP
- S_SCREEN_VIEW
- S_UK_ATTJOIN
- S_LIST_COL_INTL
- S_USER_KEY_ATT
- S_VIEW_WTMPL_IT
- S_CONTROL

- S_CONTROL
- S_EIM_ATT_MAP

Running the Repository Preparation Wizard on Release 6.2.1 Repository

6.x upgrades only. This topic applies to upgrades from Release 6.2.1.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Before you run the repository merge on an ASCII database, you must run the Repository Preparation Wizard on the Prior Customer Repository. This utility is invoked from Siebel Tools and is used to perform the following procedures:

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

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

- **Merge labels and fields.** Release 7.x merges labels with controls so that the label is now recognized as the caption property of the control. Before you run the repository merge, you need to run the Repository Preparation Wizard to merge labels, controls and fields from the previous version based upon form applet layout.

NOTE: If you are upgrading to a Siebel Financial Services 7.x application, do not use the default input file to merge labels and fields. See the Caution note in [Step 3 on page 278](#).

- **Merge Web templates.** Release 7.x merges Web templates in Siebel Tools to only use Base and Edit/Query/New for the majority of the preconfigured applets. Previous releases of Siebel used four separate templates to deliver the same functionality. Before you run the repository merge, you need to run the Merge Templates utility to merge your applet web templates.

The Repository Preparation Wizard prompts you to confirm that you completed the following steps before it proceeds with the repository merge.

NOTE: If your upgrade fails while you are preparing the prior customer repository for merge, you need to restore your database to the most recent backup and complete repository preparation procedures. If your upgrade fails after you successfully completed the “Merge labels and fields” part of repository preparation, you must restore your database to the point after which you merged labels and fields.

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

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

Choose Prior Customer Repository as the repository that you wish to prepare.

The String Migration window appears.

2 In the String Migration window:

- a** Select the language for which you wish to migrate strings.
- b** To log migrated strings, click in the check box beside Log migrated strings, then click 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 476642.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Alert 426.

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

3 In the Merge Labels and Fields window:

- a** Specify the input file by clicking the Browse button.

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.

CAUTION: If you are upgrading to a Siebel Financial Services 7.x application, you should not use the default input file, applets.txt, because the repository preparation utility operates on every form applet except those specified by the input file, and Siebel Financial Services applications contain additional applets that should not be merged. Instead, you should browse to the \Tools\Bin directory and select the input file named fi ns_applets.txt.

- b** Specify the location of Web templates.

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

- In the Merge Applet Web Templates window, click the Browse button to specify the same input file that you used in [Step 3 on page 278](#), then click Next.

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

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

The wizard prepares your Prior Customer Repository for the merge.

The Repository Preparation Wizard records the results of the labels and fields merge and the template merge into the `l abel merge. txt` file and the `templ atemerge. txt` file, respectively. If you want to view information about how the Repository Preparation wizard prepared your repository for merge, you can check these files, located in the `Si ebel_ Tool s\temp` directory. More information about these log files is provided in the table below.

File Name	Description
<code>l abel merge. txt</code>	Generated by label and field merge. Lists all applets that are being modified. For each applet modified, displays whether the label and field merge was successful. If a control already has a caption, the old caption is logged for that control.
<code>templ atemerge. txt</code>	Generated by the template merge. Lists all applets that are being modified and displays whether the merge was successful. For each applet, displays the following: <ul style="list-style-type: none"> ■ The template being used as the source (other templates are merged into this template). ■ The Applet Web Templates being inactivated. ■ Controls whose types are changed to make sure that they only appear in the correct modes.

The following errors in the `templ atemerge. txt` file are acceptable if they are against standard applets, because there were no Edit or New applet web templates associated with standard applets in the previous version.

- Upgrading APPLET_NAME.
- Using Edit as source template.
- Could not find source template.

Performing a Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

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

There are three basic categories of object differences:

- New
- Deleted
- Modified

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

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

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

Objects of the following types are deleted from the New Customer Repository:

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

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

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

A conflict occurs, however, if an object attribute was altered in both the Prior Customer Repository and the New Siebel Repository, in which case the values in all three repositories would be different. In this event, the repository merge process uses the setting of the object attribute's StandardWins flag to determine how to resolve the conflict. If this is set to Y, the attribute value from the New Siebel Repository is used; if this is set to N, the attribute value from the Prior Customer Repository is used. Conflict resolutions can be overridden for each object attribute in the New Customer Repository. See ["Reviewing Attribute Conflicts in the Repository Merge" on page 291](#) for examples of conflict resolution.

About the Repository Merge

The configuration utility that you ran while upgrading your development environment loaded two version 7.8 standard repositories. You must now use Siebel Tools to merge your existing custom configuration into one of these new repositories, creating a version 7.8 custom configuration that includes all of your previous configuration changes.

The four repositories that currently exist in your development database are listed in the following table.

Repository Name	Description
Prior V6.2.1 Siebel Repository or Prior V7.x Siebel Repository	Standard version 6.2.1 or 7.x repository, depending on the version from which you are upgrading.
Prior Customer Repository	Customized version 6.2.1 or 7.x repository, depending on the version from which you are upgrading.
New Siebel Repository	Newly loaded version 7.8 standard repository.
New Customer Repository	Newly loaded version 7.8 repository into which your custom configuration is merged.

Follow the guidelines provided in [“Optimizing Performance of the Repository Merge” on page 282](#) to improve performance of the repository merge.

The repository merge is a memory-intensive process that fails if insufficient memory is available on the Siebel Tools workstation. Before beginning a repository merge, make sure that the following preparations have been completed on the developer workstation. Make sure that the developer workstation on which Siebel Tools is running has been upgraded to the newest available version.

- Use a workstation with a minimum of 512 megabytes (MB) of RAM and at least 2 GB of virtual memory or a 2-GB page file. Otherwise, the repository merge does not complete successfully.
 - Close all other applications before performing a repository merge.
- NOTE:** Some repositories may require additional memory, based on level of customizations.
- If necessary, increase the swap space, using the Control Panel System applet, and then restart the development workstation before proceeding.
 - Ensure you allocate plenty of 32k sortwork space
 - Make sure you have a high-performing network connection.

NOTE: A slow network connection significantly increases the time required for the repository merge.

The method you use to perform a repository merge depends on whether your database uses an ASCII or EBCDIC encoding scheme:

- For ASCII databases, perform the procedure, [“Merging the Repositories for an ASCII Database” on page 283](#).

- For EBCDIC databases, perform the procedure, “[Merging Repositories for an EBCDIC Database](#)” on page 286.

Optimizing Performance of the Repository Merge

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

- 1 Optimize the machine on which you are running the repository merge.
 - Use a workstation with a minimum of 512 megabytes (MB) of RAM.
 - Allocate at least 2 GB of virtual memory, and a 2 GB page file. If the amount of virtual memory on the system is too low, performance degrades significantly.
 - Close all other applications.
 - Close all services.
 - Defragment the disk. Fragmentation significantly affects system performance.
 - On the workstation, check that the environment variable `SI EBEL_LOG_EVENTS` is set to zero. To check, enter the following command at the MS DOS prompt: `echo %SI EBEL_LOG_EVENTS%` If this variable is not set, no action is required. However, if `SI EBEL_LOG_EVENTS` is returned with a value other than zero, you must set it to zero by performing the following steps:
 - Close Siebel Tools and any other Siebel client applications.
 - Navigate to Start > Settings > Control Panel > System > Environment.
 - In the Environment dialog box, in the System Variables box, select `SI EBEL_LOG_EVENTS`. Enter 0 in the Value box, and click Set. Click OK.
 - Relaunch Siebel Tools. The new setting becomes active.

NOTE: The steps you need to take to set this variable may vary depending on the operating system you are using.
- 2 Optimize your database, because database performance can cause the repository merge to slow down considerably.
 - Make sure that temporary tablespace has enough space allocated.
 - Make sure the database has enough space allocated.
 - Make sure that the top-most logging applet in tools has no extra rows (from previous repository merge runs) when starting the repository merge.
 - Make sure that the database is not loaded with users when repository merge is run (no other users should be connected).
 - Delete extra repositories from the database using Siebel Tools. Running the repository merge on a database with more than the four repositories which are needed for the repository merge degrades repository merge performance. Before deleting extra repositories, make backups.

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

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

To increase the foreground performance of the repository merge

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

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

Merging the Repositories for an ASCII Database

Perform the following task to merge the development repositories for an ASCII database.

CAUTION: This procedure does not support EBCDIC databases. If you are upgrading a DB2 database that uses an EBCDIC encoding scheme, see [“Merging Repositories for an EBCDIC Database”](#) on page 286.

To merge the repository for an ASCII database

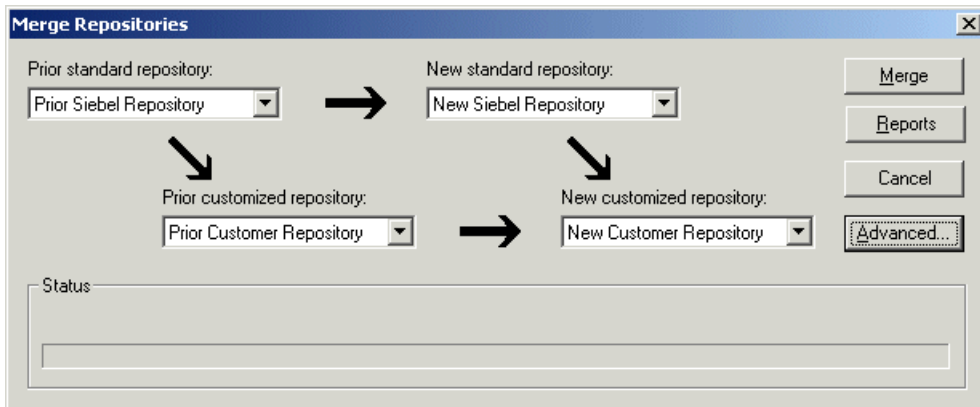
- 1 Log in to Siebel Tools and navigate to View > Options > Language Settings.
- 2 Verify that the language mode setting is set as desired.

This is the user interface language for SRF files compiled from the New Customer Repository. It will also be the language used by the postmerge utilities.

- 3 Use the File > Open Repository command to open the Prior Customer Repository.

CAUTION: Open the Prior Customer Repository, not another repository. Later steps in the repository merge process fail if you open the wrong repository.

- 4 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 Release 6.2.1 only).** This button manually launches the postmerge reports for user interface configuration described in *Upgrade Guide*. For upgrades from Release 7.x, these reports are automatically generated after the repository merge.
CAUTION: Do not select Reports unless you are upgrading from Release 6.2.1 and have 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 in [Step 5](#).
- 5 In the Merge Repositories dialog box, choose the appropriate repository name from each picklist, using the repository names listed in the following table.

Drop-Down List Item	Value to Choose
Prior Standard Repository	Prior 6.2.1 Siebel Repository or Prior 7.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

- 6 Review the settings in the Merge Repositories dialog box, and then click Advanced.

The Merge Options dialog box appears.

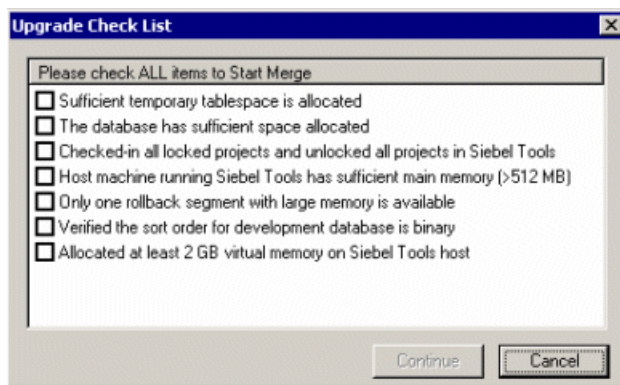
- 7 In the Merge Options dialog box, click *Abort merge if more than x errors occur*, as desired. This option aborts the repository merge automatically if more than a designated number of errors occur.

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.

- 8 To continue, click OK.

The Upgrade Check List dialog box appears.

- 9 In the Upgrade Check List dialog box, you must confirm that your environment meets the requirements for a successful repository merge. Review each requirement and select the check box if your configuration meets or exceeds the requirement.



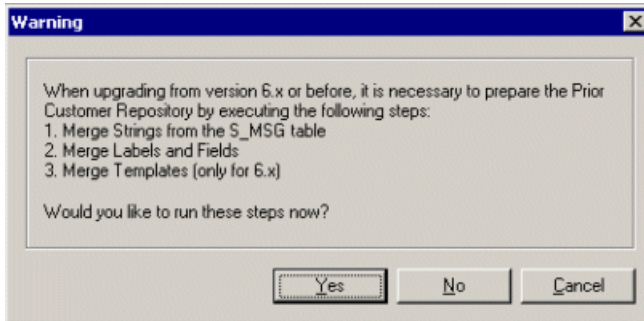
- 10 To continue, click Continue.

A warning screen prompts you to prepare the Prior Customer Repository for the merge if you are upgrading from Release 6.2.1.

- 11 In the Warning screen, choose the option for your upgrade path:

- For upgrades from Release 6.2.1, click Yes to execute Prior Customer Repository preparation steps.

- For upgrades from Release 7.x, click No to skip the Prior Customer Repository preparation steps. You completed these steps during a prior upgrade to Siebel 7.



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

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

- 12 After the merge completes, a dialog displays requesting that you make a backup of the New Customer Repository. Back up the New Customer Repository and click OK in the dialog box.

This launches the postmerge utilities. The postmerge utilities resolve common, merge-related user interface issues. See [“Running the Postmerge Utilities” on page 329](#) for more information about the postmerge utilities.

- 13 After the postmerge utilities complete, do the following:
 - Click OK in the Postmerge Utilities dialog box.
 - Close the Merge Options dialog box.

CAUTION: Do not click Merge in the Merge Options dialog box. This will restart the merge.

When the postmerge utilities start, the Postmerge Utilities dialog box displays the log entries from the utilities. The steps for reviewing the log are part of evaluating the UI and are covered in a later process. The postmerge utilities can require several hours to complete.

Merging Repositories for an EBCDIC Database

Perform the following task to complete a repository merge for an EBCDIC database.

To perform a repository merge for an EBCDIC database

1 Use the Import/Export Repository option in the Database Server Configuration Utility to export the following repositories from your prior EBCDIC database:

- Prior Standard Repository
- Prior Customized Repository
- New Standard Repository
- New Customer Repository

See *Using Siebel Tools* for information on exporting repositories using the Database Server Configuration utility.

2 Prepare a new Siebel 7.8 ASCII database on which you will perform the repository merge.

In the storage control file, the following tables must be defined with CLOBS = Yes:

- S_SCHMST_DBSCPT
- S_BI TMAP_DATA
- S_SERVI CE_SCRIPT

3 Use the Import/Export Repository option in the Database Server Configuration Utility to import the repositories exported from the EBCDIC database in [Step 1](#) into the ASCII database prepared in [Step 2](#).

NOTE: Make sure you select the Import Custom Repository option to import all languages used.

See *Using Siebel Tools* for information on importing repositories using the Database Server Configuration Utility.

4 Launch Siebel Tools against the ASCII database.

5 Run the repository merge using the procedure, [“Merging the Repositories for an ASCII Database” on page 283](#).

6 Perform the task [“Generating EIM Temporary Columns After a Repository Merge” on page 294](#).

7 Perform the task [“Determining If a Repository Merge Was Successful” on page 288](#) to review the repository merge results. Verify that the repository merge was successful, and that all reported validation messages are either acceptable or fixed.

8 Use the Database Server Configuration utility to export the merged repository (the New Customer Repository) from the ASCII database.

9 Rename the existing New Customer Repository in the EBCDIC database.

10 Use the Database Server Configuration utility to import the merged New Customer Repository back into the EBCDIC database.

11 From the Tools application pointing to the ASCII database, click on *Repository* in the Object Explorer and copy the value against the Comments column for the New Customer Repository.

- 12 Connect to the EBCDIC database through the DB2 command line and update the Comments column (with the copied value) on the table S_REPOSITORY for name= "New Customer Repository".

For example, if you copied a *Comments* value of:

```
APPLIED_PATCHES: Grid, UI NavUpgrade, MVGUpgPatch77, UI NavUpgrade, PCLWebTempl Swap, WFD, PM7.8; UpgEimCol ,
```

Then, you need to execute the following command against the EBCDIC database:

```
Update s_repository set comments
```

```
= ' APPLIED_PATCHES: Grid, UI NavUpgrade, MVGUpgPatch77, UI NavUpgrade, PCLWebTempl Swap, WFD, PM7.8; UpgEimCol '
```

Where name=New Customer Repository'

Determining If a Repository Merge Was Successful

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

The repository 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 Bridging 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.

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.
```

Each time you run the merge process, the name of the merge0.txt file is incremented, for example merge1.txt.

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.
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 ! (exclamation point).
- 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 successful.
 - If the log contains unacceptable errors, the merge has failed.
- 10 If the merge contains unacceptable errors, go to My Oracle Support and search for 477269.1 (Article ID) on My Oracle Support. This document was previously published as Siebel 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.

Migrating Custom Workflows

Upgrades: Releases 7.7.x, 7.8.x.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

Custom workflows are not migrated to the New Customer Repository during the upgprep. You must use Siebel Tools to migrate these workflows to the New Customer Repository.

The migration process has the following characteristics:

- Migrate custom workflows to the New Customer Repository during the development environment upgrade and before the upgphys. This ensures that the repository definition exported during upgphys is complete. If you migrate workflows after the upgphys, you must re-export the repository definition.

- All workflows that are in the Prior Customer Repository but not the New Customer Repository are copied to the New Customer Repository. Siebel version 0 workflows are not copied.
- If you have modified a Siebel 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.
NOTE: The Siebel Tools version must be 7.8.x or later.
- 2 Select Tools > Upgrade > Migrate Custom Workflow Processes.
The Select Repositories dialog box appears.
- 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"> ■ Prior Customer Repository. ■ Or select the repository that you are using as the Prior Customer Repository.
New customized repository	<ul style="list-style-type: none"> ■ New Customer Repository. ■ Or select the repository you are using as the New Customer Repository.

The Migrate Workflow Process dialog appears. This dialog shows 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](#) again.
- 5 Review the log file for errors.
The log file is located at `Tools_install_dir\log\upgwf.log`.

NOTE: To locate logged errors, search the log for the word *error*.

Troubleshooting

If the 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.

Reviewing Attribute Conflicts in the Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

You can change how attribute conflicts were resolved during the repository merge.

How Attribute Conflicts Occur

Attributes, such as color and height, are defined on repository objects. An attribute conflict occurs in the repository merge when all of the following are true:

- The object and attribute exist in the Prior Siebel Repository, the Prior Customer Repository, and the New Siebel Repository.
- You have changed an attribute value in the Prior Customer Repository. (The attribute value in the Prior Customer Repository and the Prior Siebel Repository are not the same.)
- The attribute has been changed in the New Siebel Repository. (The attribute in the New Siebel Repository and the Prior Siebel Repository are not the same.)

If the values in the Prior Siebel Repository and New Siebel Repository are the same but the value in the Prior Customer Repository is different, there is no conflict. The value in the Prior Customer Repository is preserved in the merged repository.

The merge process resolves attribute conflicts by referring to the setting of the Standard Win property. For about 90% of repository objects, the merge process resolves attribute conflicts by using the attribute values in the New Siebel Repository. Do not change the setting of this property.

Application Upgrade Attribute List View

You can review and change how attribute 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 on which the attribute is defined.
- **Attribute.** The attribute name.
- **Conflict.** The merge process puts a check mark in this field if there was an attribute conflict during the merge.
- **Resolution.** Displays which attribute value the merge process used to resolve the conflict:
 - **Standard Value.** The attribute value in the New Siebel Repository was used. This value is displayed in the New Standard column.
 - **Custom Value.** The attribute 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 attribute conflict is resolved. Overriding the resolution changes the attribute 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 attribute in the Prior Siebel Repository.
- **Prior Customized.** Displays the value of the attribute in the Prior Customer Repository. In the Resolution column, this value is called the Custom Value.
- **New Standard.** Displays the value of the attribute 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 [“Determining If a Repository Merge Was Successful”](#) on page 288.

To review attribute 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 attributes 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 attribute in the merged repository.

Avoid overriding conflicts for the following attributes. Visually review these attribute changes in the upgraded application before changing them:

- Left
- right
- top
- height
- width

Reviewing Deleted Objects in the Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Deleted objects are those that were in the repository when it was installed but that you have since deleted. The merge retrieves these objects from the Prior Siebel Repository and adds them to the New Customer Repository. Typically, adding these objects does not cause a problem with your upgraded configuration.

Prerequisite: The repository merge must have been successful. See [“Determining If a Repository Merge Was Successful” on page 288.](#)

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.

Reviewing Obsolete Objects in the Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 [“Determining If a Repository Merge Was Successful” on page 288.](#)

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.

Generating EIM Temporary Columns After a Repository Merge

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The repository merge process does not preserve EIM processing columns for custom mappings. After the completion of a successful merge, you must generate the missing custom EIM processing columns in the New Customer Repository.

Prerequisite: The repository merge must have been successful. See [“Determining If a Repository Merge Was Successful” on page 288.](#)

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 UpgEimCol.

This indicates that the EIM temporary columns were created successfully.

Regenerating the Repository Definition Files

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

If you have done any development after the development environment upgrade (upgphys) and before upgrading the production environment, use this topic to regenerate the schema.ddl and custrep.dat files.

The development environment upgphys upgrade step creates two repository definition files:

- **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 Siebel LogDir\schedict.log /n  
"Siebel Repository" /t dci r
```

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 IBM DB2 UDB. 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 logs 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 "New Customer 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 IBM DB2 UDB.
- *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 logs for errors. If the log indicates there are errors, contact Siebel Technical Support.

Deleting Unneeded Repository Files

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When you are confident that the repository has been upgraded successfully, export the New Customer Repository and Prior Customer Repository for safekeeping. You can also delete the following repositories:

- Prior Standard Repository
- New Standard Repository

Creating a New SRF File

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

You can optionally create an .srf file immediately after the repository merge. This enables you to run an application and evaluate whether you have resolved attribute conflicts correctly.

After the you have completed the upgphys portion of the upgrade, you must create a new .srf file. This file includes the merged repository changes.

If you will review your UI in more than one language, set the Tools Language setting and compile an .srf file for each language. For information about how to compile an .srf file for a specific language, see *Global Deployment Guide*.

Prerequisite: The repository merge must have been successful. See [“Determining If a Repository Merge Was Successful” on page 288](#).

Compiling a New SRF File Immediately After the Repository Merge

Perform this optional procedure immediately after the repository merge and before you perform the upgphys upgrade in the development environment. Repeat this procedure as needed to help evaluate how you have resolved attribute conflicts.

Do not use this procedure to compile an .srf file if you have completed the upgphys upgrade. Instead, see [“Compiling a New .srf File After the Upgphys Upgrade” on page 298](#).

To compile a new .srf file after the repository merge

- 1 Navigate to the following directory on an upgraded Siebel Server you will use to test applications in the development environment. Verify that this is *not* the directory where Siebel Tools is installed:

Windows: `SI EBEL_ROOT\OBJECTS\language`

UNIX: `$SI EBEL_ROOT/OBJECTS/language`

Where:

language is the installed language, for example, ENU.

- 2 Rename the `siebel.srf` file to `siebel_release.srf`.
- 3 On the Siebel Tools development workstation, connect to the database against which you ran the merge.
- 4 Start Siebel Tools.
- 5 Choose File > Open Repository.
- 6 Select New Customer Repository.
- 7 Choose Tools > Compile Projects.
- 8 Select All Projects.
- 9 Enter the path to the Siebel Server OBJECTS directory where you renamed the `siebel.srf` file, and specify `siebel.srf` as the file name.

CAUTION: Do *not* enter the path to the Siebel Tools OBJECTS directory. You will overwrite and destroy the Siebel Tools `.srf` file.

- 10 Click Compile.
- 11 Copy the contents of the Siebel Tools WEBTEMPL directory to the Siebel Server WEBTEMPL directory.
- 12 Start the application.

Compiling a New .srf File After the Upghys Upgrade

After completing the upghys upgrade in the development environment, you must compile a new `.srf` file. This updates the file with the merged repository changes. This is the `.srf` file you will copy to the Siebel Servers in your test environment.

To compile a new .srf file after the upgphys upgrade

- 1 Navigate to the following directory on an upgraded Siebel Server you will use to test applications. Verify that this is not the directory where Siebel Tools is installed:

Windows: *SIEBEL_ROOT\OBJECTS\language*

UNIX: *\$SIEBEL_ROOT/OBJECTS/language*

Where:

language is the installed language, for example, ENU.

- 2 Rename the *si_ebel.srf* file to *si_ebel_release.srf*.
- 3 On the Siebel Tools development workstation, connect to the database against which you ran the merge.
- 4 Start Siebel Tools.
- 5 Choose File > Open Repository.
- 6 Select Siebel Repository.
- 7 Choose Tools > Compile Projects.
- 8 Select All Projects.
- 9 Enter the path to the Siebel Server OBJECTS directory where you renamed the *si_ebel.srf* file, and specify *si_ebel_release.srf* as the file name.

CAUTION: Do *not* enter the path to the Siebel Tools OBJECTS directory. You will overwrite and destroy the Siebel Tools *.srf* file.

- 10 Click Compile.

Siebel Tools creates a new *.srf* file and places it at the specified location on the Siebel Server.

16 Migrating Siebel 6.2.1 Customizations

If you are upgrading your development environment from Release 6.2.1, you must carry out the tasks in this chapter. This chapter contains the following topics:

- [“About Migrating to the Siebel Web Client”](#)
- [“About Migrating 6.2.1 Scripts” on page 307](#)
- [“About Migrating 6.2.1 Client-Side Interfaces” on page 308](#)
- [“Identifying What Will Be Migrated” on page 309](#)
- [“Exposing Hidden Properties of Applets and Views” on page 310](#)
- [“Running the Siebel Web Client Migration Wizard” on page 311](#)
- [“Reviewing Applet and View Migration” on page 314](#)
- [“Running the Web Layout Wizard” on page 315](#)
- [“Migrating Scripts Attached to Controls” on page 316](#)
- [“Migrating Scripts Attached to Applets” on page 318](#)
- [“Migrating Business Component, Business Service, and Application Scripts” on page 321](#)
- [“Migrating Outbound COM Interfaces” on page 322](#)
- [“Migrating Inbound COM Interfaces” on page 323](#)
- [“Regenerating the Postmerge Utilities Log” on page 326](#)
- [“Reviewing Customized Business Components” on page 326](#)

About Migrating to the Siebel Web Client

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

In Release 7.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, and does not require additional persistent software installed on the client. The browser connects through a Web server to the Siebel Server, which executes business logic and accesses data from the Siebel Database. Only the user interface layer of the Siebel Business applications architecture resides on the user computer.

The Siebel 7 Web Client Migration Wizard upgrades customized view and applet definitions used by the Siebel Dedicated Client, Windows Thin Client, and Java Thin Client in previous Siebel releases to the new Siebel Web Client. You run this wizard after the repository has been upgraded.

The Siebel Web Client Migration Wizard performs the following functions:

- Recognize what types of applets are migrated
- Provide a model applet for each applet type
- Select a template automatically based on the model applet
- Populate the model applet automatically
- Migrate applets and views from Siebel 6.2.1 customer applications

The Release 7.8 Siebel Web Client migration results in a grid-based layout for form applets. The grid-based layout allows you to control the layout of a form applet using the Web Layout Editor. You can drag and drop controls without having to modify the Web templates. Siebel Web Client Migration of form applets using grid-based functionality preserves applet paradigms, similar to the Release 6.2.1 dynamic grid.

The Siebel Web Client Migration Wizard does not migrate Siebel VB or Siebel eScript scripts attached to applets. You must either move these scripts to the server or rewrite them in JavaScript, the Siebel Web Client-supported language for user interface scripting.

The Release 7.x Siebel Web Client supports most of the events and methods on user interface objects supported by previous Siebel releases. In addition, the Siebel Web Client provides comparable inbound and outbound integration capabilities 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.

For detailed information about the grid layout, see *Configuring Siebel Business Applications*.

Example of the Siebel Web Client UI

Figure 6 shows a standard Siebel 6.2.1 Windows Client. Figure 7 on page 304 shows a view that has been migrated from the Windows Client to the Siebel Web Client. This view is running in a browser.

New	Last Name	First Name	Mr/Ms	Call	Work Phone #	Work Fax #	Home Phone #
>	Alan	Wiley	Mr.		(425) 810-9988	(425) 810-9706	
	Alder	Kevin	Mr.		(415) 491-2360 x334	(415) 491-2200	
	Allen	Donna	Ms.	✓	(415) 717-2002	(415) 717-9999	(415) 247-3323
	Alterman	David	Mr.		(510) 452-2000	(510) 452-2000	
	Alvarez	Harlan	Mr.		(510) 223-9320	(510) 223-2323	
	Atkins	Charlie	Mr.	✓	(510) 223-2319	(510) 223-2328	
	Bass	Ken	Mr.		(415) 329-1000	(415) 329-1001	
	Basset	Mary	Ms.		(510) 292-0945	(512) 292-6544	
	Bell	Jeff	Mr.		(415) 717-2009	(415) 717-9999	
	Bell	Robert			(704) 992-3487		
	Bingenheimer	David	Mr.		(925) 485-5439	(510) 594-6111	(925) 640-6021
	Bolt	Maria	Ms.		(815) 435-8899	(815) 435-8800	
	Brianic	Byrne	M		(510) 782-1016	(510) 782-1023	
	Brighton	Amy	Mrs.		(415) 654-9872	(415) 654-1111	
	Brown	Scott	Mr.	✓	(510) 567-2300	(510) 567-2222	
	Burson	Robert	Mr.		(404) 493-2400	(404) 493-2401	
	Campos	Miguel	Mr.		(510) 292-1228	(512) 292-6544	
	Choi	Victor					
	Delgado	Abel	Mr.				(415) 234-5847
	Douglas	John	Mr.		(510) 292-1002	(512) 292-6544	
	Finkel	Michelle	Ms.		(617) 254-8255	(617) 254-5002	
	Floyd	Sara	Ms.		(415) 717-2056	(415) 717-9999	
	Franklin	Bill	Mr.	✓	(212) 733-9000	(212) 733-9110	

Figure 6. Contact List View—Windows Client

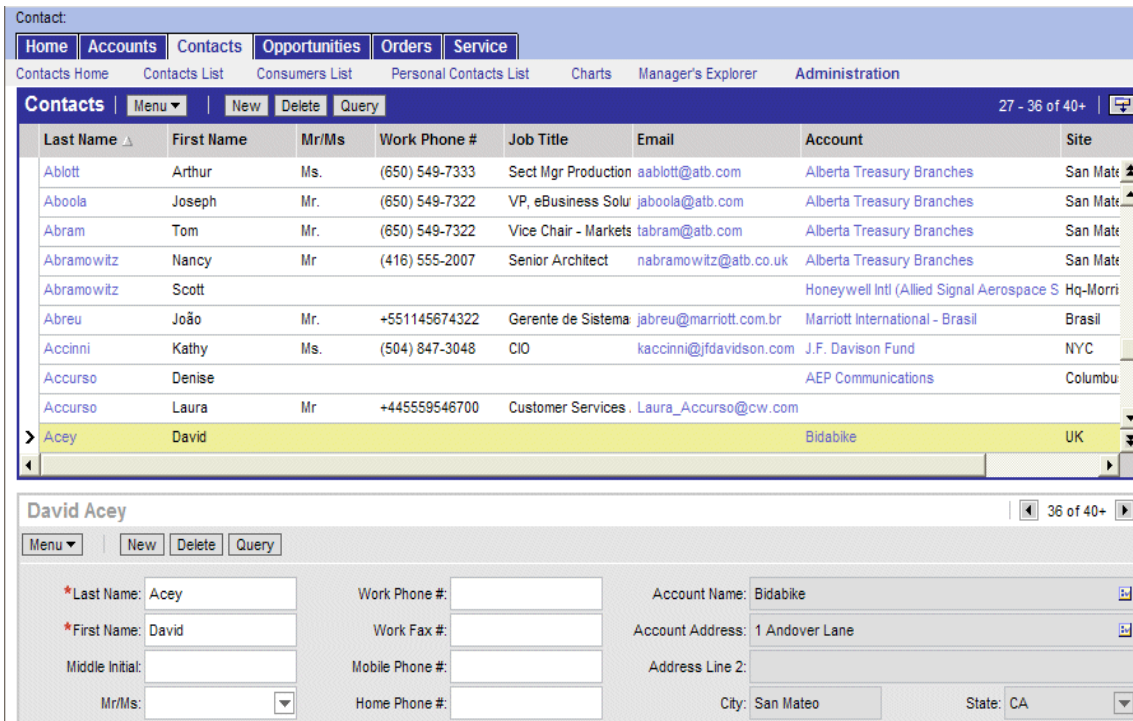


Figure 7. Contact List View—Siebel Web Client

About the Siebel Web Client Migration Wizard

When you upgrade to Release 7.8, 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 Siebel 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 Siebel 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 Siebel 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 Siebel 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 Siebel 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 to migrate your applets and views is time-consuming and is not recommended.

Model Applets and Views

The Release 7.8 repository contains model applets and views. The Siebel 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 27](#) lists preconfigured model applets and views available in Release 7.8.

Table 27. 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 27. 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 Siebel 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 Siebel 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 6.2.1 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, as described in [Table 28](#).

Table 28. 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 awtcvtcfg.txt file (located in the binary subdirectory of the root Siebel Tools installation directory), the applet is not converted to grid layout.

About Migrating 6.2.1 Scripts

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

You must migrate scripts on UI objects to server scripts or browser scripts when upgrading to Release 7.8.

Server Scripts

Server scripts execute in the Application Object Manager. They existed in prior releases and continue to be supported. 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 Siebel 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 476879.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 386. For more information about events that are exposed through the DOM, see *Siebel Tools Online Help*.

You must migrate Release 6.2.1 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 Release 6.2.1 applet scripts such as Control_Click event handlers to business components
- Rescripting Release 6.2.1 ActiveX controls in browser script
- Rewriting other Release 6.2.1 applet scripts in browser script

To identify the number of 6.2.1 applet scripts to be reviewed or migrated, look at the Object List Editor in Flat mode in Siebel Tools.

You must review Release 6.2.1 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 Release 6.2.1 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*.

About Migrating 6.2.1 Client-Side Interfaces

Upgrades: Release 6.2.1 only.

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 Siebel 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 Release 6.2.1, desktop applications communicated with the Siebel Client through an Automation interface. In Release 7.8.x, the high interactivity Siebel 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, Siebel Mobile Web Client, and Siebel Dedicated Web Client. (Unless otherwise specified, references in this section to the Siebel client may refer to any or all of these clients.)

Identifying What Will Be Migrated

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This topic describes how to use the Siebel Web Client Migration Wizard to identify the created and modified applets and views that will be migrated when upgrading to Release 7.8.

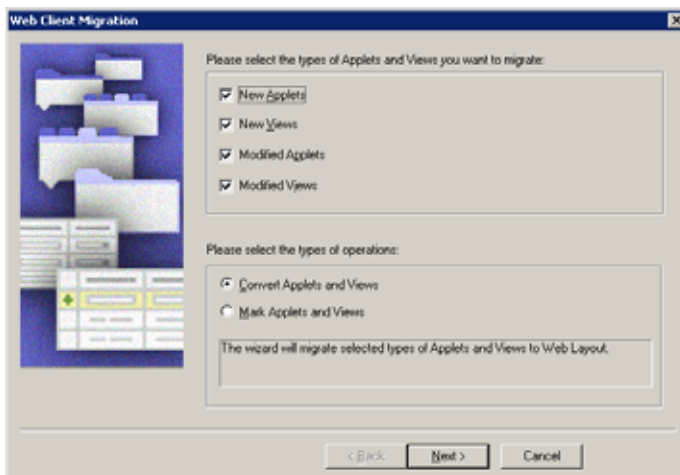
For more information on migrating created or modified applets and views to the Siebel Web Client, see [“About Migrating to the Siebel Web Client” on page 301.](#)

Prerequisite: The repository merge must have been completed successfully. See [“Determining If a Repository Merge Was Successful” on page 288.](#)

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 Siebel 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 Siebel Web Client Migration Wizard identified.

Exposing Hidden Properties of Applets and Views

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

As of Release 7.0.x, certain properties in Siebel Tools are no longer applicable for the Siebel Web Client. By default, these properties are hidden. Before you begin the migration to the Siebel 7.8 Web Client, expose these properties to make the migration of your customizations to Release 7.8 easier.

For more information on migrating created or modified applets and views to the Siebel Web Client, see [“About Migrating to the Siebel Web Client” on page 301.](#)

To expose hidden properties

1 Using a text editor, open the tool s.cfg file located in the following directory:

```
Siebel Tool sDir\bin\LANG
```

2 Under the [Siebel] section in the tool s.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 tool s.cfg file.

5 Launch Siebel Tools.

The previously hidden properties are now visible.

- 6 After completing the Siebel Web Client migration to Release 7.x, reset the `ClientConfigurationMode` parameter to `Web` to hide properties that are no longer applicable.

Running the Siebel Web Client Migration Wizard

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This topic describes how to use the Siebel Web Client Migration Wizard to batch-migrate your new and modified applets and views to Release 7.8.

For more information on migrating created or modified applets and views to the Siebel Web Client, see [“About Migrating to the Siebel Web Client” on page 301.](#)

Prerequisites

- The repository merge must have been completed successfully. See [“Determining If a Repository Merge Was Successful” on page 288.](#)
- Identify model applets and model views. See [“Identifying What Will Be Migrated” on page 309.](#) This prerequisite is optional.
- Expose hidden properties on objects. See [“Exposing Hidden Properties of Applets and Views” on page 310.](#)

Related Topic

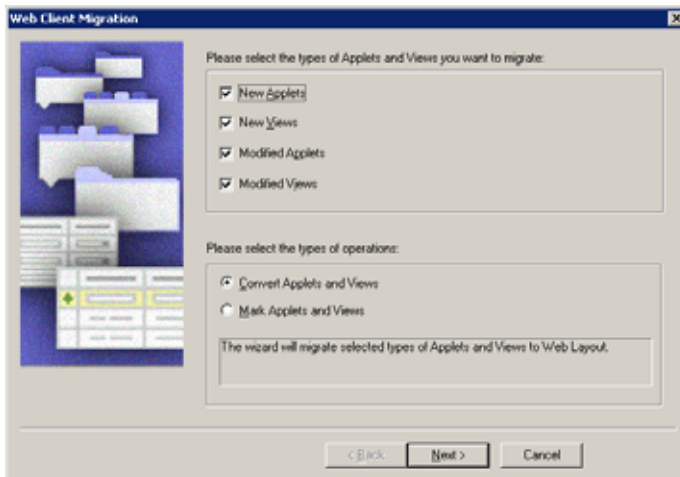
[“Running the Web Layout Wizard” on page 315](#)

To run the Siebel 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.

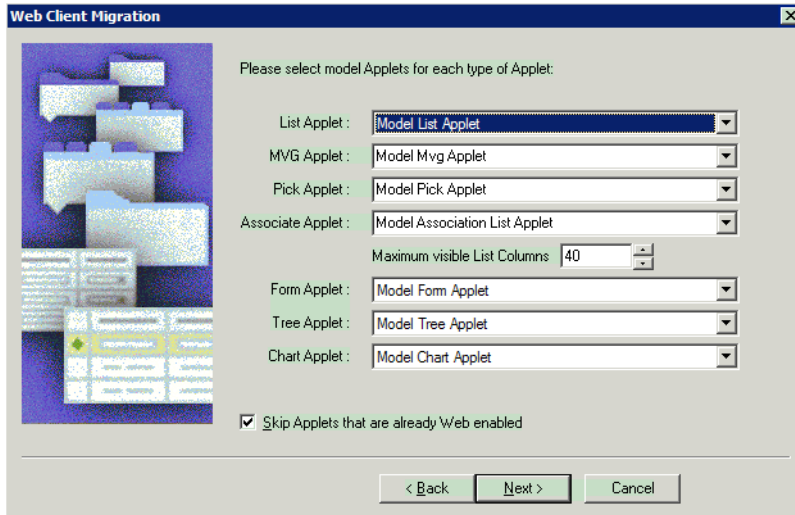
This starts the Siebel Web Client Migration Wizard. The model applet selection screen appears.



- 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.

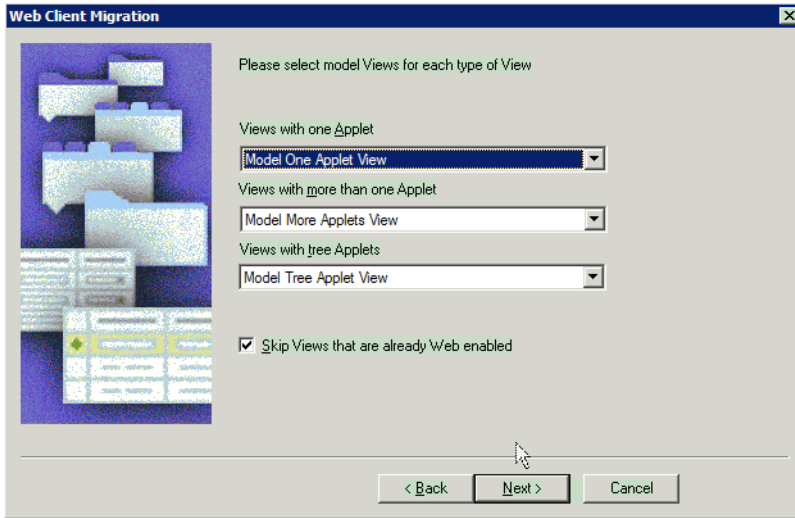
To continue, click Next.

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 Siebel 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 Applet and View Migration

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The Siebel 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.

For more information on migrating created or modified applets and views to the Siebel Web Client, see [“About Migrating to the Siebel Web Client” on page 301](#).

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.

Running the Web Layout Wizard

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If some views or applets did not migrate properly when you ran the Siebel Web Client Migration Wizard to upgrade to Release 7.8, you can remigrate them using the Web Layout Wizard. With the Web Layout Wizard you can migrate applets or views individually or in groups.

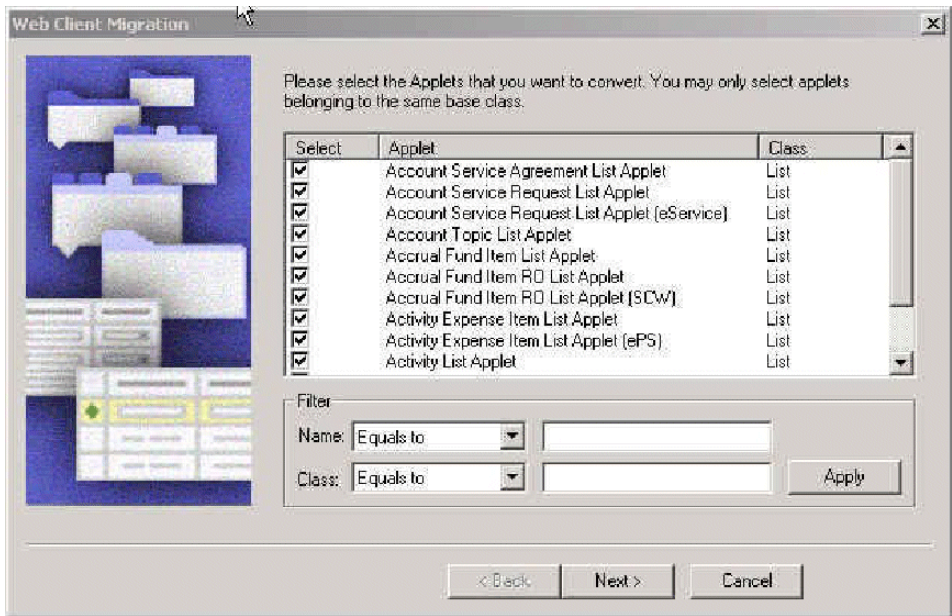
When you select applets for migration, they must be in the same base class. When you select views, they must be of the same type.

You can run the Web Layout Wizard on applets and views multiple times.

For more information on migrating created or modified applets and views to the Siebel Web Client, see [“About Migrating to the Siebel Web Client” on page 301](#).

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 box, 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.

Migrating Scripts Attached to Controls

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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 Release 7.x, all control methods are available through browser scripting and execute in the high interactivity applications only. Table 29 provides a list of the control methods that are supported by browser script. For more information, see *Siebel Tools Online Help*.

Table 29. 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 8 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.

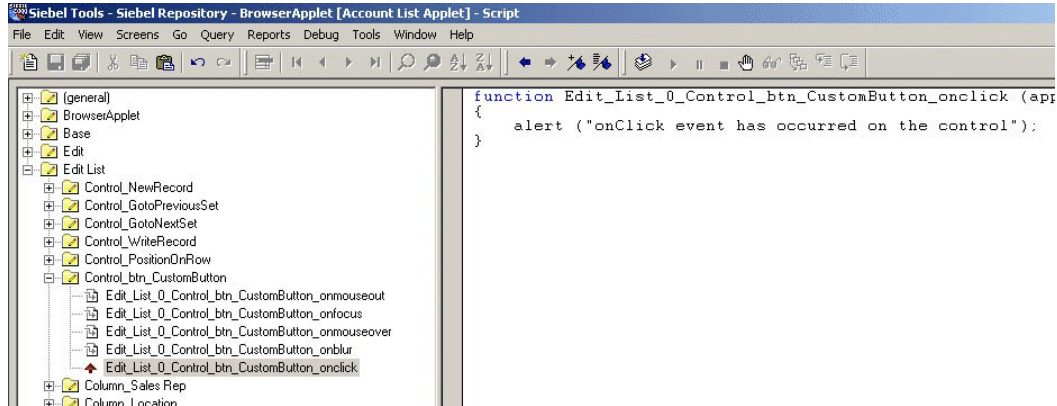


Figure 8. DOM Events When MiniButton Control Added to Applet

For more information on migrating scripts, see [“About Migrating 6.2.1 Scripts”](#) on page 307.

Related Topics

[“Migrating Scripts Attached to Applets”](#)

[“Migrating Business Component, Business Service, and Application Scripts”](#) on page 321

Migrating Scripts Attached to Applets

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When upgrading to Release 7.8, you must do one of the following:

- Migrate Release 6.2.1 scripts written on applets to applet server scripts or business component server scripts. Migrating applet scripts includes the following types of tasks:
 - Moving Release 6.2.1 applet scripts such as Control_Click event handlers to business components
 - Rescripting Release 6.2.1 ActiveX controls in browser script
 - Rewriting other Release 6.2.1 applet scripts in browser script
- Rewrite Release 6.2.1 scripts as browser scripts.

To identify the number of 6.2.1 applet scripts to be reviewed or migrated, look at the Object List Editor in Flat mode in Siebel Tools.

As shown in Figure 9, applet scripts implemented in Siebel 6.2.1 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

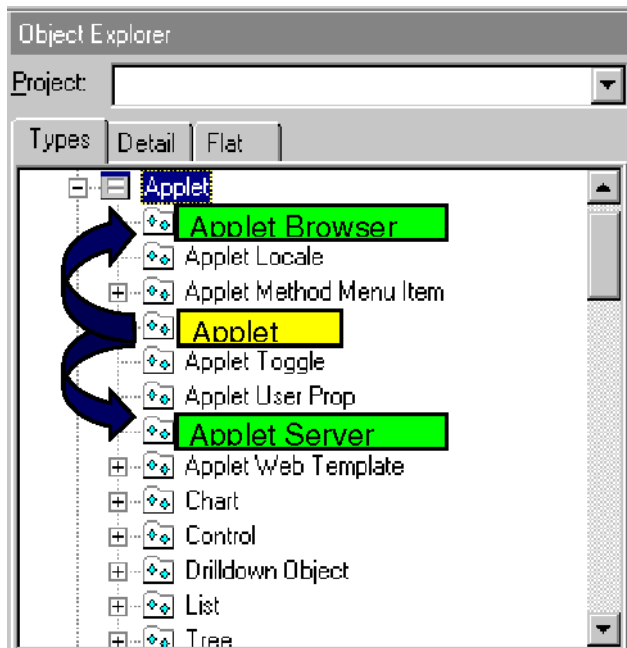


Figure 9. Movement of Applet Scripts

In Release 7.x, the WebApplet_ShowControl and WebApplet_ShowListColumn event handlers are only supported for standard interactivity applications.

In many cases, the migration of applet scripts is straightforward, because several WebApplet events correspond to Siebel 6.2.1 applet events. Corresponding browser or server script event handlers might be available, but the scripts are not automatically migrated. Table 30 depicts Siebel 6.2.1 events and corresponding Release 7.x events available in either browser or server script.

Table 30. Siebel 6.2.1 Applet Events and Release 7.x Equivalents

Siebel 6.2.1 Applet Event	Release 7.x Equivalent
Applet_Load	WebApplet_Load (Server Event)
Applet_GotFocus	WebApplet_Load (Server Event)
Applet_ChangeFieldValue	Applet_ChangeFieldValue (Browser Event)
Applet_ChangeRecord	Applet_ChangeRecord (Browser Event)
Applet_PreInvokeMethod	WebApplet_PreInvokeMethod (Browser or Server Event)
Applet_InvokeMethod	WebApplet_InvokeMethod (Browser or Server Event)

In cases where there is no direct correlation between Siebel 6.2.1 events and Release 7.8 events (for example Applet_LostFocus event), you must evaluate and reimplement the functionality of the Siebel 6.2.1 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 Release 7.x. The obsolete applet methods are listed in Table 31. 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 31. Obsolete Applet Methods

Applet Method
ActiveControl
FindControl
FindActiveXControl
GotoControl
PostChanges
Drilldown

For more information on migrating scripts, see [“About Migrating 6.2.1 Scripts”](#) on page 307.

Related Topics

[“Migrating Scripts Attached to Controls”](#) on page 316

[“Migrating Business Component, Business Service, and Application Scripts”](#)

Migrating Business Component, Business Service, and Application Scripts

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The majority of application, business components, and business services scripts implemented in Siebel 6.2.1 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 Release 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, Release 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 Release 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 476612.1 (Article ID) on My Oracle Support. This document was previously published as Siebel FAQ 1562.

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 Siebel 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 Release 7.x, see *Siebel Object Interfaces Reference* and 476879.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 386. For more information on migrating scripts, see [“About Migrating 6.2.1 Scripts” on page 307](#).

Related Topics

[“Migrating Scripts Attached to Controls” on page 316](#)

[“Migrating Scripts Attached to Applets” on page 318](#)

Migrating Outbound COM Interfaces

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 Excel App;  
  
var Sheet;  
  
Excel App = new Acti veXObj ect("Excel . Appl i cati on");  
  
Sheet = new Acti veXObj ect("Excel . Sheet");  
  
// Make Excel vi si ble  
  
Sheet. Appl i cati on. Vi si ble = true;  
  
// Pl ace 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)
 - 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*.

For more information on migrating client-side interfaces, see [“About Migrating 6.2.1 Client-Side Interfaces” on page 308](#).

Related Topic

[“Migrating Inbound COM Interfaces”](#)

Migrating Inbound COM Interfaces

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Release 7.x external applications can interact with the Siebel client through the Siebel Web Client Automation Server. The Siebel Web Client Automation Server is similar in functionality to the COM Automation Server available in Siebel 6.2.1, but it does not expose applet or control objects.

The Release 7.x Siebel 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 Siebel Web Client Automation Server is implemented as a COM object resident within the browser.

To enable the Siebel 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 Siebel 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 Siebel Mobile Web Client Automation Server or the Siebel Web Client Automation Server, see *Siebel Tools Online Help*.

You might need to modify browser security settings to allow use of the Siebel Web Client Automation Server. For security settings and information about deploying ActiveX controls in the browser, see *Siebel System Administration Guide*.

For more information on migrating client-side interfaces, see [“About Migrating 6.2.1 Client-Side Interfaces” on page 308](#).

Related Topic

[“Migrating Outbound COM Interfaces” on page 322](#)

Regenerating the Postmerge Utilities Log

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When upgrading from Release 6.2.1 to Release 7.8, you perform the repository merge and then the Siebel Web Client migration. This invalidates part of the postmerge utilities log created after the repository merge. You must regenerate the postmerge utilities log before using it to evaluate the merge.

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 Siebel Web Client Migration Wizard.

To regenerate the postmerge utilities log

- 1 In Siebel Tools, 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.

Reviewing Customized Business Components

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After upgrading the Siebel Database to Release 7.8, 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

Upgrades from Release 7.0.x and 7.5.x

The table `S_EMPLOYEE` is migrated to three tables—`S_CONTACT`, `S_EMP_PER`, and `S_USER`. After you have determined and implemented the approach for previously defined custom extension columns on obsolete tables, you should manually configure the business component field to reference that database column. Perform any manual reconfiguration in the New Customer Repository after the upgrade has been run.

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*.

17 Postmerge Development Tasks

This chapter lists the tasks you perform after you have completed the repository merge during a development environment upgrade. It includes the following topics:

- “Running the Postmerge Utilities”
- “Troubleshooting Postmerge UI Problems” on page 330
- “Verifying Business Address Applet Configuration” on page 331
- “Reviewing Grid-Based Applets After the Merge” on page 332
- “Reviewing UI Navigation After the Merge” on page 333
- “Reviewing Multi-Value Group (MVG) Shuttle Applets After the Merge” on page 334
- “Revising Rich Text Controls After the Merge” on page 336
- “Reviewing New Aggregate Categories After the Merge” on page 337
- “Revising Visibility Filters to Display Correctly After the Merge” on page 338
- “Assigning a Category and Type to Chart Views After the Merge” on page 339
- “Assigning a Category and Type to Explorer Views After the Merge” on page 340
- “Setting Up Navigation to Inaccessible Detail Views After the Merge” on page 340
- “Eliminating Obsolete UI Fields After the Merge” on page 341
- “Reviewing UI Objects Affected by ICL After the Merge” on page 342
- “Reviewing Required Fields in the UI After Merge” on page 342
- “Setting Label Alignment for Text Fields” on page 343
- “Assigning an Item Identifier to Web Template Items” on page 343

Running the Postmerge Utilities

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After the merge is completed successfully, run the postmerge utilities. These utilities make the following changes to user interface objects in the merged repository:

- Convert flow-based form applets to grid-based.
- Verify that new or customized screens, views, and applets from the Prior Customer Repository are configured correctly and can be accessed in the UI.

- Revise multi-value group (MVG) applets so that they are shuttle-enabled.
- If you selected Incorporate Custom Layout (ICL) during the repository merge, the utilities list the UI objects that were affected.
- Lists UI objects that have missing required fields.

You can rerun the postmerge utilities as needed. If the postmerge utilities encounter a problem, do the following:

- Try to fix the problem in the merged repository, and then rerun the utilities.
- If you cannot fix the problem, restore the backup of the postmerge Siebel Database or repository. Then, resolve the problem and run the utilities again.

The postmerge utilities take a significant amount of time to run.

Prerequisite. Before running the postmerge utilities, verify you have a backup of the Siebel Database or the Siebel Repository that was done after the repository merge.

To run the postmerge utilities

- 1 If you are rerunning the postmerge utilities and want to save the existing log, rename the log. The path to the log is as follows:

```
SIEBEL_TOOLS_INSTALL_DIRreppatch\log\reputility.log
```

If you do not rename the log, it will be overwritten.

- 2 In Siebel Tools, navigate to Screens > Application Upgrader > Application Upgrade Object List.
- 3 In the Application Upgrades list, select the record of the merge.
- 4 In the Application Upgrades list, right-click and select Launch Post Merge utility from the pop-up menu.

A dialog box appears and displays the postmerge utilities log. When the utilities have finished, a message showing completion displays in the log.

Troubleshooting Postmerge UI Problems

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

Basic Troubleshooting Guidelines

If views or screens do not display, do the following:

- Verify that the user has been assigned the correct responsibilities.
- 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 Release 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`

Verifying Business Address Applet Configuration

Upgrades: Release 7.0.x only. This topic applies only to Siebel Industry application (SIA) upgrades. It does not apply to Siebel Business 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 Grid-Based Applets After the Merge

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

Reputility log section: “Start Invalid Web Template Item Mapping Clean-up.”

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The postmerge utilities convert many form applets to grid-based layout in Release 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, many form applets are converted 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. For more information on Incorporate Custom Layouts (ICL), see [“About the Incorporate Custom Layout \(ICL\) Upgrade Option” on page 264.](#)

The reputility.log, 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\reputility.log`

[Table 32](#) lists common issues and corrective actions for grid-based applets.

Table 32. 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. Action: 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. Action: 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. Action: None required.

Table 32. Common Grid-Based Applet Issues

Log Examples	What To Do
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. Action: 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. Action: 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. Action: 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. Action: 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. Action: Review the applet and verify that the button control displays correctly.

Reviewing UI Navigation After the Merge

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Development environment only.

Reputility log section: "User Interface Navigation Upgrade."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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
```

Table 33 shows examples of common issues and corrective actions.

Table 33. 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. Action: 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. Action: Remove the invalid screen view record that references the invalid view.
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. Action: 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. Action: Verify that the screen view category exists, and then check database access.

Reviewing Multi-Value Group (MVG) Shuttle Applets After the Merge

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

Reputility log section: "Multivalued Group Shuttle Applet Upgrade."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

As of Release 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`

Table 34 lists common issues and corrective actions for MVG applets.

Table 34. Common MVG Applet Issues

Log Entry Example	What To Do
WARNING:: [APPLET: Account Address Mvg Applet (NB)] MVG Applet is Inactive. Ignoring Applet.	This applet is inactive. Action: No action required.
WARNING:: [APPLET: Primary Employee Mvg Applet] [Applet Web Template: Base] Applet Web Template is Inactive. Ignoring Applet Web Template.	The applet Web template is inactive. Action: No action required.
WARNING:: [APPLET: FINS Application Contact Mvg Applet - ACAPS] [CONTROL METHOD INVOKED: ExecuteQuery] Has a Non Standard Control Type.	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. Action: Revise the control definition and map it to the Edit List mode.
WARNING:: [APPLET: State Model - State Mvg Applet] [CONTROL METHOD INVOKED: UndoQuery] Has a Non Standard Control Type.	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. Action: Revise the control's definition and map it to the Edit List mode.
WARNING:: [APPLET: LOY Account Address Assoc Applet] [CONTROL: CancelQuery] Control is Inactive. Please inspect and Reactivate.	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. Action: Redefine a Cancel button and map it to the Edit List mode.
WARNING:: [APPLET: LOY Account Address Assoc Applet] [CONTROL: ExecuteQuery] Control is Inactive. Please inspect and Reactivate.	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. Action: Redefine a Go button and map it to the Edit List mode.

Table 34. 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>Action: 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>Action: 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 Release 7.7, the UI standard is that when MVGs first display, they are editable.</p> <p>Action: 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>Action: Move the control at location 107 to another location, and then map the Go button to location 107.</p>

Revising Rich Text Controls After the Merge

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

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 4, "How to Perform the Upgrade."](#)

As of Release 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
```

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

Reviewing New Aggregate Categories After the Merge

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

Reputility log section: “Issue 2: New Aggregate Category Records That Should Be Renamed.”

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The following new properties were added to the Screen View object in Siebel Tools to support new UI navigation introduced in Release 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
```

Revising Visibility Filters to Display Correctly After the Merge

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

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 4, “How to Perform the Upgrade.”](#)

In Release 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 views 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
```

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 one of the applets to 1.
- 4 Refine the query to display the Parent Category listed in the log.

Which applet has the Item Identifier set to 1 is not important. Verify that only one applet has the Item Identifier set to 1.

Assigning a Category and Type to Chart Views After the Merge

Upgrades: Release 7.x not using ICL.

Environments: Development environment only.

Reputility log section: "Issue 4: Chart Views Needing Migration to Aggregate Type."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

The Chart menu item is relocated as of Release 7.7 and displays in the applet header. 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 in a nonstandard position in the applet header.

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
```

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.

Assigning a Category and Type to Explorer Views After the Merge

Upgrades: Release 7.x not using ICL.

Environments: Development environment only.

Reputility log section: “Issue 5: Explorer Views Needing Migration to Aggregate Type.”

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

As of Release 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
```

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.

Setting Up Navigation to Inaccessible Detail Views After the Merge

Upgrades: Release 7.x not using ICL.

Environments: Development environment only.

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 4, “How to Perform the 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, 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
```

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.

Eliminating Obsolete UI Fields After the Merge

Upgrades: Release 7.x using ICL.

Environments: Development environment only.

Reputility log section: "Issue 7: Fields deprecated from Business Components."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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
```

Reviewing UI Objects Affected by ICL After the Merge

Upgrades: Release 7.x using ICL.

Environments: Development environment only.

Reputility log section: "Issue 8: List of the items affected by PCL."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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
```

Reviewing Required Fields in the UI After Merge

Upgrades: Release 7.x using ICL.

Environments: Development environment only.

Reputility.log section: "Issue 9: List of Required Fields Missing from the UI."

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the 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 repositivity 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\repositivity.log
```

Setting Label Alignment for Text Fields

Upgrades: Release 7.x using ICL.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

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 set label alignment for text fields

- 1 Using a text editor, open the main.css file in the Siebel Tools installation directory:

```
Windows: \publi c\</ang>\files\main.css
```

```
UNIX: /publi c//ang/files/main.css
```

where

lang is the installed language, for example enu.

- 2 Search for mceLabel2.

This is a section containing several entries. Verify that the section begins .mceLabel2, not .mceLabel.

- 3 In the .mceLabel2 section, set the following values as shown:

- vertical-align : top

- text-align: left

- font-weight : bold

- 4 Save the file and exit.

Assigning an Item Identifier to Web Template Items

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

All View Web Template items must have an item identifier. 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.

18 Postupgrade Tasks for a Development Environment

This chapter describes the tasks you perform after upgrading your development environment. It includes the following topics:

- [“Reapplying Custom Columns to the Siebel Database”](#)
- [“Deleting Duplicate EIM Mappings” on page 347](#)
- [“Regenerating the Database Template File” on page 348](#)
- [“Extracting Developers or Siebel Tools Clients” on page 349](#)
- [“Resetting Database Server Configuration Parameters” on page 349](#)
- [“Upgrading to the Symbolic String Model” on page 349](#)

Reapplying Custom Columns to the Siebel Database

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Several tables used in prior versions of Siebel Business applications are no longer used, or have been replaced by new tables, in the current release. If you added extension columns or foreign key (FK) columns to tables that are obsolete in the current release, you may want to reapply these changes to the new tables that have replaced them.

The upgrade process generates a report that you can review for information about obsolete tables that will help you decide whether or not you need to reapply schema customizations. This report, `xtndobstbl.txt`, lists the following:

- Obsolete tables in the current release
- Custom columns in obsolete tables
- Custom foreign key columns pointing to obsolete tables
- EIM mappings for custom foreign key columns to obsolete tables
- Workflow columns by custom foreign key to obsolete tables
- Custom denormalized columns to Siebel base tables that might be obsolete

Each obsolete table is listed with one of three status codes:

- **Not Used.** These tables are not used in the current release but you can continue to use them; for example, `S_ORG_PROD` or `S_PERIOD_BU`. These tables are supported as is (for instance, with docking or EIM).

- **EOL (end of life).** These tables are not used in the current release, and it is recommended that you delete them because they will not be supported in future releases; for example, old S_*_IF tables.
- **Inactive.** These tables have been discontinued, and are not supported in the current release. Extension columns that reside on inactive tables must be moved to alternate tables; for example, S_EMPLOYEE should be moved to S_CONTACT, S_USER, or S_EMP_PER.

If no tables are listed in xtndobstbl.txt, no action is required. If this file lists any tables, you can reapply their custom extensions and foreign key columns to tables in the current release using Siebel Tools. For further information on this task, see *Configuring Siebel Business Applications*.

Table 35 lists examples of previously used tables which are inactive in Release 7 (you can no longer use these tables) and lists the suggested new tables to which custom extensions need to be reapplied. The new tables are recommendations only; the tables that you need to apply the extensions to might vary depending on their type and use. Contact Siebel Technical Services to validate the reapplication of extension columns and to review the steps necessary to migrate any extension column data to the new tables.

This data must be migrated during both the development and production environment upgrades.

Table 35. Examples of Tables That Are Obsolete in Release 7

Previous Table	Suggested New Table
S_EMPLOYEE	S_CONTACT, S_USER, S_EMP_PER
S_EMP_POSTN	S_PARTY_PER
S_ORG_INT	S_ORG_EXT, S_BU
S_POSTN_RPT_REL	S_PARTY_RPT_REL

If you have created many custom extension columns on the tables S_EMPLOYEE or S_ORG_INT, both of which are no longer used in Release 7, the joins between the tables will not be accurate. This may result in SQL errors when you launch the Siebel client.

In such cases, using Siebel Tools, you need to manually create corresponding extension columns in the new target tables, and manually move the data to the new extension column on the new table before you continue migration of the application. You should then review the business component configuration to make sure that the client will operate properly.

You may need to do this in one of the following instances:

- Fields based on custom extension columns in S_EMPLOYEE or S_ORG_EXT
- Fields based on custom extension tables from S_EMPLOYEE or S_ORG_INT with or without join
- Custom joins to custom extension tables from S_EMPLOYEE or S_ORG_INT

If you review the xtndobstbl.txt file after you run the buscomp migration utility, you will find a list of fields that require your attention.

Table 36 lists examples of previously used tables that are no longer used in Release 7, but that you may want to continue to use.

Table 36. Examples of Tables That Are Not Used in Release 7

Previous Table	Suggested New Table
S_CRSE	S_SRC, S_SRC_EVT
S_CRSE_OFFR	S_SRC, S_SRC_EVT
S_CRSE_REG	S_SRC_REG
S_CTLG_CAT_REL	S_CTLG_CAT
S_OPTY_PROD	S_REVN
S_TMSHT_LINE	S_TMSHT_ITEM, S_TMSHT_LN

Table 37 lists examples of tables which were unused in previous releases of Siebel Business applications, but are now used in Release 7.

Table 37. Examples of Previously Unused Tables That Are Used in Release 7

Now Used Table	Used to be...
S_ACT_EMP	S_EVT_ACT
S_ACT_CON	S_EVT_ACT

Deleting Duplicate EIM Mappings

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

Any custom Enterprise Integration Manager (EIM) mappings that were added to *_IF tables in previous versions of Siebel Business applications are automatically migrated to *_IF tables during the upgrade to the current release. However, those customizations are not migrated to the EIM tables that were introduced in Siebel 7.x to replace the *_IF tables. If you choose to use the new EIM tables rather than the *_IF tables, you must manually add any customizations to them.

If custom mappings in *_IF tables are duplicated by new mappings added as part of the current release of Siebel Business applications, errors will result when you use EIM.

The Database Server Configuration Wizard that you ran to upgrade the physical schema generated a file called mapclash.txt. This file lists any custom EIM mappings that conflict with Siebel-provided mappings. The file contains the columns illustrated and defined in [Table 38](#).

Table 38. Mapclash.txt File Column Names and Definitions

Column Name	Definition
IT_Name	Interface table name (example: S_ACCOUNT_I F)
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 Prior 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_I F, 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.

Regenerating the Database Template File

Upgrades: All upgrades.

Environments: Development environment only.

Following the upgrade, you must regenerate the SQL Anywhere template database file used by Siebel Remote. This process updates its schema to the same version as the Siebel Database Server. You will use the Generate New Database component from a new Siebel Server to do this.

For procedures on regenerating the SQL Anywhere Template file, refer to *Siebel Remote and Replication Manager Administration Guide*.

Extracting Developers or Siebel Tools Clients

Upgrades: All upgrades.

Environments: Development environment only.

Extract all Siebel Tools developers and clients using the Database Extract component from a new Siebel Server.

After you have extracted and initialized the mobile databases of all Siebel Tools developers, they must check out a read-only copy of all projects into the local database. For further information on setting up Siebel Tools developers, see *Using Siebel Tools*.

Testing the Upgraded Configuration

You must test the upgraded configuration thoroughly to make sure that all functionality works as expected. You need to execute your full development or acceptance test plan, including at least one synchronization session from a mobile client, before proceeding to upgrade your production environment.

If you make further changes to your customized configuration, remember to re-export the repository to the `custrep.dat` file before running the production environment upgrade.

After you have thoroughly and successfully tested your upgraded configuration, your development environment upgrade is complete. Proceed with [Chapter 10, “Preparing a Production Environment for Upgrade.”](#)

NOTE: Save the log files generated during the repository upgrade and schema upgrade. Locate the `custrep.dat` file in the `DBSRVR_ROOT\COMMON` subdirectory. Save the `custrep.dat` file for your production upgrade.

Resetting Database Server Configuration Parameters

Upgrades: All upgrades.

Environments: Development environment only.

After you complete your upgrade, you need to reset your Siebel Database Server configuration to installation settings. See *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

Upgrading to the Symbolic String Model

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Release 7.7 introduced a symbolic string model which replaced 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.

Procedures for converting or consolidating to the symbolic strings model are found in *Using Siebel Tools*.

19 Postupgrade Tasks for a Production Environment

This chapter lists the tasks you perform after upgrading your production environment. It includes the following topics:

- “Upgrading Siebel Mobile and Dedicated Web Clients”
- “Upgrading Regional Servers”

Upgrading Siebel Mobile and Dedicated Web Clients

Upgrades: All upgrades.

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

After a successful upgrade of Siebel Business applications, you are ready to upgrade your Siebel Mobile and Dedicated Web Clients.

To upgrade Siebel client software on Siebel Dedicated Web Clients and Siebel Mobile Client workstations

- 1 Uninstall the previous version of the Siebel client application.
- 2 Install the Release 7.8 version of the Siebel client application. You can use either of the following methods for the installation:
 - Use the Siebel Image Creator utility to uncompress JAR files and create a Siebel network image, or staging area, for your Siebel applications.
 - Use a custom installer created using the Siebel Packager utility. For more information about the Packager utility, see *Implementing Siebel Business Applications on DB2 UDB for z/OS*.

Upgrading Regional Servers

Upgrades: All upgrades.

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

After the production upgrade has executed successfully and the system integrity has been adequately verified, database extracts must be run for all Regional Servers in the deployment. Before initialization of the Regional Nodes, you must drop all Siebel tables, views, and indexes using the appropriate documentation from the RDBMS vendor. See *Siebel Remote and Replication Manager Administration Guide* for more information on database extracts, and regional server initialization.

20 Postupgrade Tasks for Database and File System

This chapter describes the database and file system tasks you carry out after upgrading to Release 7.8. It contains the following topics:

- [“Checking for Inactivated EIM Table Columns”](#)
- [“Validating Dock Objects and Rule Definitions”](#)
- [“Updating the File System Directory” on page 354](#)
- [“Updating File System Attachments” on page 355](#)

Checking for Inactivated EIM Table Columns

Upgrades: Releases 7.0.x, 7.5.x.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

In Release 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 application function.

Validating Dock Objects and Rule Definitions

Upgrades: All upgrades.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If you deploy Siebel Business applications to mobile users with local databases, you can run the DICTUTL 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*.

NOTE: Changes to visibility rules and dock objects require the assistance of Siebel Technical Support or Siebel Expert Services.

To verify that 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 39](#):

```
dictutl /C ODBC_DATASOURCE /U USERNAME /P PASSWORD /D TABLEOWNER /N
"REPOSITORY_NAME" /A y 2> logfile.log
```

Table 39. Command Line Flags for DICTUTL

Flag	Parameter	Description	Required
/C	<i>ODBC_DATASOURCE</i>	ODBC datasource name	Yes
/U	<i>USERNAME</i>	User name to log in to database	Yes
/P	<i>PASSWORD</i>	User password to log in to database	Yes
/D	<i>TABLEOWNER</i>	User name of tableowner	Yes
/N	<i>"REPOSITORY_NAME"</i>	Name of repository for dictionary (the parameter must be bounded within double quotes)	Yes
/A	<i>y or n</i>	Enter the <i>y</i> parameter to ignore the dictionary cache. Enter <i>n</i> if you do not want to ignore the dictionary cache.	Yes

Review the `LOGFILE.log` file.

Related Topic

["Preserving Dock Objects and Visibility Rules" on page 117](#)

Updating the File System Directory

Upgrades: Release 6.2.1 only.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

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 in the Siebel File System and copy file system attachments to it.

To make file attachments accessible to Siebel Anywhere

- 1 In the existing file system structure (for example C:\siebfile), create an att subdirectory if it does not already exist.
 Windows example: C:\siebfile\att
 Creating the att subdirectory does not adversely affect the installation of your Siebel Server. (You install the Siebel Server at a later point.)
- 2 Copy (do not move) all files located under the \siebfile directory to the \siebfile\att directory so that all file attachments are accessible by Siebel Business applications.
- 3 Verify that files have copied correctly to the \siebfile\att directory.
 After this has been verified, clean up the file system.

Updating File System Attachments

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Since 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 sections below to update Siebel File System attachments.

Updating Attachment File Names

NOTE: The task in this section applies to all upgrades.

Siebel 7.8 provides a utility to update attachment file names in the Siebel File System. [Table 40](#) 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 40. Input Arguments for the Update Utility

Upgrade Path	Old Table	New Table
Upgrades: Release 6.2.1 only.	■ S_EMPLOYEE_ATT	■ S_CONTACT_ATT

Table 40. Input Arguments for the Update Utility

Upgrade Path	Old Table	New Table
Upgrades: Release 6.2.1 only.	■ S_ORG_INT_ATT	■ S_ACCNT_ATT
■ Upgrades from: Siebel 6.2.1 FINS.	■ S_INSCLM_BL_ATT	■ S_INVOICE_ATT
■ Upgrades from: Siebel 7.0.x, 7.5.x, 7.7.x, & 7.8.x. Siebel Insurance only.		

To update file system attachments

- 1 Navigate to the following directory:

Windows: `SIEBEL_ROOT\bin`

UNIX: `$SIEBEL_ROOT/bin`

- 2 Enter the following command:

Windows: `chg_file_sys.bat OLD_TABLE NEW_TABLE "FILE_SYSTEM"`

UNIX: `chg_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` = name of the new table to which the original data was migrated.
- `"FILE_SYSTEM"` = name of the directory where the Siebel File System attachments reside (entered inside quotation marks).

Windows example:

```
chg_file_sys.bat S_EMPLOYEE_ATT S_CONTACT_ATT
"C:\siebfile\att"
```

UNIX example:

```
chg_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`, you need to rename a file such as `S_EMPLOYEE_12-1ABC.SAF` to `S_CONTACT_12-1ABC.SAF`.

Updating Attachments for S_LIT

NOTE: Perform the task in this section if you are upgrading from Siebel release 6.x and 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`, so that the files correspond to the new table name.

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
- `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:\sea78\siebsrvr"  
"C:\sea78\dbsrvr"
```

UNIX example:

```
file_upg_mm.ksh SEBL sadmi n sadmi npw SIEBEL /usr/siebel/siebfie $SIEBEL_ROOT  
/usr/siebel/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.

21 Postupgrade Tasks for Applications

This chapter describes the Siebel application-specific tasks you perform after upgrading to Release 7.8. It contains the following topics:

- [“Generating Reporting Relationships After Upgrade” on page 360](#)
- [“Setting Up Your Environment to Support Global Time Zone” on page 362](#)
- [“Updating Enterprise Application Integration \(EAI\) After Upgrade” on page 363](#)
- [“Setting Visibility Modes for Access Control” on page 364](#)
- [“Removing Call Center Duplicate Logins After Upgrade” on page 367](#)

Siebel ERM

- [“Upgrading ERM Customized Microsite and Group News Pages” on page 367](#)
- [“Migrating Course Duration Information for Siebel Training” on page 369](#)
- [“Upgrading the Launch Field in Siebel Training LOV” on page 369](#)
- [“Upgrading the Test Status in Siebel Training After Upgrade” on page 370](#)
- [“Verifying Class and Session Times in Siebel Training” on page 370](#)
- [“Upgrading ERM Approval Business Process Workflows” on page 371](#)

Siebel Financial Services

- [“Migrating Data to the Bankruptcy Status Field” on page 373](#)
- [“Upgrading File System Attachments for Siebel Financial Services Call Reports” on page 373](#)
- [“Upgrading File System for Household Notes and Attachments” on page 374](#)

Siebel Marketing

- [“Upgrading Marketing Responsibilities” on page 376](#)
- [“Reviewing Marketing Campaign Data” on page 376](#)
- [“Reviewing Renamed Fields in Siebel Marketing” on page 377](#)
- [“Displaying Marketing Regions” on page 377](#)
- [“Revising Marketing Program Flowchart Icons” on page 378](#)
- [“Setting Default Campaign Execution Options” on page 378](#)
- [“Upgrading Activity Plans for Programs and Campaigns” on page 378](#)
- [“Upgrading the Newsletter Offer Type” on page 379](#)
- [“Configuring Universal Inbox” on page 379](#)

- “Reviewing Data Warehouse Schema Changes” on page 380

Order Management

- “Upgrading Siebel Purchase Orders” on page 380
- “Configuring Asset-Based Ordering” on page 381
- “Reviewing Address Data After Upgrade” on page 382
- “Configuring Products and Quotes” on page 383

Siebel Pricer

- “Upgrading Attribute Pricing” on page 386
- “Verifying the Upgrade to Aggregate Discounts in Pricer” on page 386

Financial Services

- “Migrating Data to the Bankruptcy Status Field” on page 387

Workflows

- “Upgrading Seeded Workflows” on page 388
- “Upgrading Inbound Workflows” on page 388

Product Configurator

- “Upgrading Siebel Product Configurator” on page 389

Generating Reporting Relationships After Upgrade

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

The Generate Reporting Relationships process needs to be executed after the upgrade to Release 7.8 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 Release 7.8, 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 Release 7.x configuration includes the Generate Reporting Relationships feature as a hidden button on the Position List Applet NB. You need to go through Siebel Tools configuration to expose this button.

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 Siebel 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 administrator. The administrator needs to 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 Your Environment to Support Global Time Zone

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If you previously upgraded to Release 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 Release 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 Release 7.x, it is strongly recommended that you operate your production environment with global time zone enabled.

See [“Upgrading UTC Delta Columns” on page 363](#) for information about upgrading 7.x data for columns that were *not* UTC-enabled in Release 7.x that changed to UTC-enabled in the current release.

Enabling Global Time Zone

To enable global time zone support, after your upgrade is complete, you need to set the global time zone parameter (Universal Time Coordinated system preference) to TRUE through Siebel Tools. The UTC system preference is not enabled after an upgrade to Release 7.x. To enable the global time zone feature, you must run the UTC conversion utility. The high-level steps you need to perform are outlined in the following procedure (see *Global Deployment Guide*).

To enable global time zone support after an upgrade

- 1 Stop Siebel Servers.
- 2 Set UTC system preference to FALSE before you convert your historical data.
- 3 Prepare your data for conversion to global time zone, as instructed in *Global Deployment Guide*.

- 4 Convert your historical data, using the UTC conversion utility, to make all existing date/time values consistent with global time zone logic. Perform this step as instructed in *Global Deployment Guide*.
- 5 After you have confirmed that your UTC conversion was successful, turn on global time zone by setting the UTC system preference to TRUE. To reset this parameter through Server Manager, navigate to Application Administration > System Preferences.
- 6 Bring up Siebel Servers and Web Server.

Upgrading UTC Delta Columns

If you enabled UTC time zone support in a previous upgrade, perform the following steps to upgrade the delta columns (those that are newly UTC-enabled in the current release).

For more information about UTC, see *Global Deployment Guide*.

To upgrade delta columns to UTC

- 1 Verify that you have a current backup of your database.
- 2 Open the master_utc.ucf. It is located in the following directory in the Siebel Database Server installation:
 - Windows: `DBSRVR_ROOT\DATABASE_PLATFORM\master_utc.ucf`
 - UNIX: `DBSRVR_ROOT/DATABASE_PLATFORM/master_utc.ucf`
- 3 Edit the File Name parameter by replacing `dri ver_utc.ucf` with `dri ver_utc_delta.ucf`.
- 4 Launch the appropriate Database Server Configuration utility:
 - UNIX: `dbsrvr_config.ksh`
 - Windows: Start > Programs > Siebel Enterprise Servers > Configure DB Server

You can also start the wizard under Windows by typing `ssi ncfgw -l language_code -v Y` at a Command prompt. The *language_code* is the three-letter language code, in all capitals, for the language in which you want the GUI to display—for example ENU for English.

If you are prompted to select an .scm file, select `dbsrvr.scm`.

- 5 When you reach the Database Server Options prompt, select Run Database Utilities.
- 6 At the Database Utility Selection prompt, select Universal Time Code Conversion.
- 7 Continue until you are prompted to run the Siebel Upgrade Wizard, then click OK.

Updating Enterprise Application Integration (EAI) After Upgrade

Upgrades: All upgrades.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 during upgrading to a new version of Siebel Business applications.

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 Enterprise Application Integration*.

- 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.

Setting Visibility Modes for Access Control

Upgrades: Release 6.2.1 only.

Environments: Development environment only.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Certain areas of the Release 7.8 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.

Removing Call Center Duplicate Logins After Upgrade

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

During the upgrade to Release 7.8, 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 > Administration - User.
- 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).

Upgrading ERM Customized Microsite and Group News Pages

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

As of Release 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 CSSWEFrameListERMPageRender.
- 9** Add the new virtual business component to the Business Object of 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 articles on My Oracle Support.

Migrating Course Duration Information for Siebel Training

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When upgrading from a release prior to Release 7.8, 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.

Upgrading the Launch Field in Siebel Training LOV

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 Release 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.

Upgrading the Test Status in Siebel Training After Upgrade

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

As of Release 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 *Completed-Failed*.
- If all tests have been passed at least once, mark the status as *Completed-Passed*. 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 *Completed-Passed*.
- Else *CompletedFailed* takes precedence over *Completed-Passed*.

Verifying Class and Session Times in Siebel Training

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 7.8, the way class and session start times and end times are displayed is changed. For existing classes and sessions you must do the following:

- For classes, verify that the times displayed in the Start Date and End Date fields are correct.
- For sessions, verify that the times displayed in the Start and End fields are correct.

To verify class times are correct

- 1 Navigate to Administration-Training > Class Details.
- 2 In the Classes applet, expose the following hidden fields:

- Start Time
- End Time

Alternative: These fields are displayed in the form applet.

- 3 For each class record, compare the time in the Start Time field with the time in the Start Date field.
- 4 Right-click in the Start Date field to adjust the time to be the same as that in the Start Time field as required.
- 5 Compare the time in the End Time field with the time in the End Date field.
- 6 Right-click in the End Date field to adjust the time to be the same as that in the End Time field as required.
- 7 Repeat this procedure for all the displayed classes.
- 8 Hide the Start Time and End Time fields.

To verify session times

- 1 In the sessions applet, expose the following hidden fields:
 - Start Time
 - End Time
- 2 Select a class in the Classes applet.
- 3 For each session in the Sessions applet, compare the time in the Start Time field with the time in the Start field.
- 4 Right-click in the Start field to adjust the time to be the same as that in the Start Time field as required.
- 5 Compare the time in the End Time field with the time in the End field.
- 6 Right-click in the End field to adjust the time to be the same as that in the End Time field as required.
- 7 Select the next class in the Classes applet and repeat this procedure.
- 8 When you have examined all the sessions for all the classes, hide the Start Time and End Time fields.

Upgrading ERM Approval Business Process Workflows

Upgrades: Release 7.5.x only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

In Release 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 verify conformance to data access policies at your site, see [“Configuring Universal Inbox” on page 379](#).

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.

Migrating Data to the Bankruptcy Status Field

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 7.8, a BK_STATUS_CD column is provided in the S_BANKRUPTCY table. This column stores Bankruptcy status information that is used by the Siebel Financial Services application.

If you have implemented a bankruptcy status field in the release you are upgrading from, use an SQL command to migrate the data from the custom extension column to BK_STATUS_CD.

Upgrading File System Attachments for Siebel Financial Services Call Reports

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When you upgrade to Release 7.7.x, call report attachments from S_ACTIVITY_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_ACTIVITY*.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_uni x.ksh  
/l LOGFILE_LOCATION/mv_act_to_comm_att_uni x.log SCRIPT_LOCATION/  
mv_act_to_comm_att_uni x.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_uni x.ksh

Upgrading File System for Household Notes and Attachments

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

If you are upgrading a Siebel Financial Services application, 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: `chng_file_sys.bat OLD_TABLE NEW_TABLE "FILE_SYSTEM"`

UNIX: `chng_file_sys.ksh -s OLD_TABLE -t NEW_TABLE -f "FILE_SYSTEM"`

where:

- `OLD_TABLE` = Name of the original table.
- `NEW_TABLE` = Name of the new table to which the original data was migrated.
- `"FILE_SYSTEM"` = Name of the directory where the file system resides (entered inside quotation marks).

Windows example:

```
chng_file_sys.bat S_ORGGRP_ATT S_CONTACT_ATT "C:\siebfile\att"
```

UNIX example:

```
chng_file_sys.ksh -s S_ORGGRP_ATT -t S_CONTACT_ATT
-f "/usr/siebel/siebfile/att"
```

4 Review the renamed files carefully to verify that they can be accessed by Siebel Financial Services applications.

Opportunity Product Migration

During the upgrade, 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`, that child object needs to be manually remapped to `S_REVN`.

A report generated during the upgrade identifies which columns or tables you need to examine. This report, `xtndobstbl.log`, lists extension columns that reside on obsolete tables and therefore need to be moved to alternate tables. For more information about reapplying custom extension columns on obsolete tables, see ["Checking for Inactivated EIM Table Columns" on page 353](#).

Upgrading Marketing Responsibilities

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Release 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.

Reviewing Marketing Campaign Data

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

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.

Reviewing Renamed Fields in Siebel Marketing

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Some applet fields have been renamed from previous releases. If you have written any customized user documentation, you need to update references to these fields. Renamed fields are shown in [Table 41](#).

Table 41. 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

Displaying Marketing Regions

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After the upgrade, the Region field for Marketing Plans, Programs, and Campaigns does not display. This is because the Region field is associated with the Region hierarchy rather than the Region LOV field.

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.

Revising Marketing Program Flowchart Icons

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

After the upgrade, 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*.

Setting Default Campaign Execution Options

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This postupgrade task 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*.

Upgrading Activity Plans for Programs and Campaigns

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This postupgrade task is not mandatory but may be required by local business processes.

As of Release 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.

Upgrading the Newsletter Offer Type

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

This postupgrade task is not mandatory but may be required by local business processes.

As of Release 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.

Configuring Universal Inbox

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

As of Release 7.7, Universal Inbox is implemented as a business service. It provides management of approval queues in both Siebel Employee Relationship Management (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 Release 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 Employee Relationship Management (ERM). Application administrators can see all approval tasks across all applications.

Reviewing Data Warehouse Schema Changes

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Customers who use Siebel Marketing and the Marketing Server must manually upgrade their metadata for those cases in which they have mapped to tables or fields in the Siebel main database or the Business Data Warehouse that have changed in Release 7.x. Due to the extensive changes to the Data Warehouse schema in Release 7.x, releases, entirely new mappings may be required. See *Siebel Marketing Installation and Administration Guide* for information on table and field metadata mappings.

Upgrading Siebel Purchase Orders

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

If upgrading from a release prior to Release 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.

The utility requires the following parameters:

- *USERNAME* = Siebel user login name.
- *PASSWORD* = Siebel login password.
- *LANG* = Language used.

- *CFG_FILE* = The configuration file used to launch the application.
- *DATA_SOURCE* = Date source used from the .cfg file.

To update the Transaction Amount field in Payment Lines

- Enter the following command:

```
Pmntupgd /u USERNAME /p PASSWORD /l LANG /c CFG_FILE /d DATA_SOURCE
```

Configuring Asset-Based Ordering

Upgrades: Releases 7.0.x, 7.5.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Release 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 Release 7.0.4 or Release 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 Release 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 Release 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 42](#). If you have written scripts or made other changes regarding these buttons or named methods, you must manually add these after the upgrade.

Table 42. 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 Release 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 Release 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 Releases 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 Release 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 Address Data After Upgrade

Upgrades:

- From Release 6.2.1 of Siebel Financial Services applications to Release 7.8.x of Siebel Industry applications.
- From Release 7.0.x of Siebel Financial Services applications to Release 7.8.x of Siebel Industry applications on all other platforms.
- From Release 7.8.x of Siebel Business applications to Release 7.8.x of Siebel Industry applications on all other platforms.
- 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.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

The way address data is stored changed at Release 7.8. 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 any 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.
- If a record in S_ADDR_ORG has the same value for ADDR_NAME as a record in S_ADDR_PER, the upgrade process appends the ROW_ID to ADDR_NAME in one of the records.
- 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 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 7.0.x Siebel Industry Solutions upgrades to 7.8 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/log/rpt_dup_addr_names.txt*

- 2 Use the report to identify records that are duplicates. For each set of duplicate records, the upgrade appended the ROW_ID to ADDR_NAME in one of the records.
- 3 Use queries to edit records or delete duplicate and obsolete records.

Configuring Products and Quotes

Upgrades: Release 6.2.1 only.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, "How to Perform the Upgrade."](#)

You may need to adjust or configure some customizable products and quotes after upgrade.

Customizable Products

After upgrading to Release 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 correct 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)
;
commi t
;

```

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_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_uni x.ksh
/l LOGFILE_LOCATION/mv_quote_to_order_att_uni x.log SCRIPT_LOCATION/
mv_quote_to_order_att_uni x.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
/e	Not applicable	Turn off statement printing in spool file	N
/h	Not applicable	Turn off column headers for queries	N
/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_uni x.ksh

Upgrading Attribute Pricing

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 7.8, the attribute adjustments feature replaces attribute pricing in Siebel Pricer. The attribute adjustments feature is part of the dynamic matrix pricing feature.

You must manually upgrade attribute pricing data to attribute adjustments by running a business service method. The business service method does the following:

- Upgrades attribute pricing headers to attribute adjustment headers
- Upgrades attribute pricing attributes to dynamic matrix dimensions
- Upgrades attribute pricing values to dynamic matrix dimension domains
- Upgrades attribute pricing adjustment items to dynamic matrix rules

To upgrade attribute pricing to attribute adjustments

- 1 Launch Siebel Sales.
- 2 Verify that all attribute classes have been upgraded to product classes.
- 3 From the application-level menu, choose Site Map > Administration - Business Service > Simulator.
- 4 Create a new record:

Field	Value
Service Name	PSP Pricer Upgrade
Method Name	UpgradeDynamicMatrix

- 5 Click Run to start the business method.

Verifying the Upgrade to Aggregate Discounts in Pricer

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 7.8, the aggregate discounts feature replaces bundle factors in Siebel Pricer. The bundle factor definitions are upgraded to aggregate discounts, and the sequencing of bundle factors are upgraded to aggregate discount sequences.

The name of the aggregate discount in Pricer will be set to *bundle factor name + row ID of the record*. This is because Pricer requires the aggregate discount name to be unique.

Sequencing of bundle factors within a pricing model is upgraded to aggregate discount sequencing. The name of the aggregate discount sequence is set to the pricing model name that contained the bundle factors.

The Price List and Price List Item will be stamped with the appropriate aggregate discount sequence name. In prior releases, the pricing model was specified at the Price List or, for customized products, at the Price List Line Item level. This ensured execution of the bundle factors at runtime.

At Release 7.8, the execution of aggregate discounts at runtime requires the association of the aggregate discount sequences at the Price List or Price List Line Item level.

The upgrade process makes the following assumptions about Pricer implementations prior to Release 7.8:

- Flowcharts were used to chain up bundle factors in the pricing model
- The bundle factor with the lowest sequence is connected to the "Aggregate Start" step
- Each Aggregate Start sequence contains only bundle factors and does not contain aggregate factors
- The next factor in the flowchart (when True or False) always has a larger sequence number.

If your implementation does not meet all the above criteria, the upgrade process moves the definitions to the appropriate Pricer entities (such as aggregate discounts), but the sequences will not be correct.

In such cases, you must manually verify that aggregate discount sequences chain up the aggregate discounts as intended. Use the sequence of execution that existed prior to the upgrade.

To verify upgrade to aggregate discounts

- 1 Launch Siebel Sales.
- 2 Navigate to Administration – Pricing > Aggregate Discount Sequences.
- 3 For each aggregate discount sequence, drill down to the detail view.
- 4 Locate the aggregate discount that corresponds to the first preupgrade bundle factor. Verify that it has the lowest sequence number. If not, revise the numbers in the Sequence, Next Discount If Used, and Next Discount If Not Used columns.
- 5 Verify that the numbers in the Next Discount If Used and Next Discount If Not Used columns are greater than the number in the Sequence column. Also verify that they point to the expected aggregate discounts. If not, revise the numbers in all three columns as required.

Migrating Data to the Bankruptcy Status Field

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

Environments: Production test, production.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

At Release 7.8, a BK_STATUS_CD column is provided in the S_BANKRUPTCY table. This column stores Bankruptcy status information that is used by the Siebel Financial Services application.

If you have implemented a bankruptcy status field in the release you are upgrading from, use an SQL command to migrate the data from the custom extension column to BK_STATUS_CD.

Upgrading Seeded Workflows

Upgrades: All upgrades.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

Customizations to seeded workflows were saved and migrated during upgrade, but you must manually reimplement them in order for them to work properly.

In Release 7.7, workflow definitions were relocated to the Tools Repository.

To upgrade a seeded workflow

- 1 In the Siebel Repository, revise each seeded workflow so that a new copy is created with a new version number.
- 2 Manually merge in your customizations, and then deploy and activate the workflow.

Upgrading Inbound Workflows

Upgrades: Release 7.0.x.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the 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 Release 6.2.1 to 7.8, 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.
```

Transitioning from Development to Production

To transition to production, you must import tables from the main Siebel Database, as well as importing the Siebel Tools Repository. A special utility, REPIMEXP, is provided to migrate the Siebel Tools Repository to production. Refer to the *Siebel Business Process Designer Administration Guide* for the steps required to transition to production.

The tables are named as follows:

- Siebel Database tables containing workflow information begin S_WFA.
- Siebel Tools Repository tables containing workflow information begin S_WFR.
- Siebel Database tables that contain workflow information for releases prior to Release 7.8 begin S_WF_ (Note the underscore after WF.) After the upgrade to Release 7.8, these tables are obsolete and are not referenced by applications.

If you have long-running workflows, verify they are all functioning correctly before transitioning to production.

Upgrading Siebel Product Configurator

Upgrades: Release 6.2.1 only.

Environments: All environments.

This topic is part of an upgrade process. See [Chapter 4, “How to Perform the Upgrade.”](#)

When the upgrade process is finished, you must examine your upgraded models, in the form of customizable products, and release the products for end user use.

22 Tuning the Production Upgrade Scripts

This chapter describes ways in which you can improve the performance of the production environment upgrade by tuning the production upgrade scripts in a test environment.

This chapter contains the following topics:

- [“About Tuning the Upgrade Scripts”](#)
- [“Optimizing Unload and Load Job Performance” on page 392](#)
- [“Adding the Statistics Clause to Load Cards” on page 392](#)
- [“About Deactivating Jobs That Do Not Process Data” on page 393](#)

About Tuning the Upgrade Scripts

Upgrades: All upgrades.

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

In Release 7.8, you can tune the SQL upgrade scripts in a production test environment to improve their performance and then re-use these tested scripts in the live production environment. For example, the scripts used to upgrade your Siebel Database are generic. They update your Siebel Database to support all Siebel applications' functionality. You can reduce downtime by tuning these scripts to optimize performance by eliminating unneeded SQL statements. You can then re-use these revised scripts in your production upgrade.

You can tune your production upgrade scripts at any time after upgrading the Siebel Database schema in your production test environment.

Contacting Siebel Expert Services

CAUTION: You are required to contact Siebel Expert Services for approval of any upgrade script tuning that you perform. If you do not, you may invalidate your support agreement.

It is recommended (but not required) that you contact Siebel Expert Services for help with the following tasks:

- Running load and unload jobs in parallel
- Changing the job submission order

If you want to change the submission order of jobs other than the load and unload jobs, you must first obtain approval from Siebel Expert Services because many jobs have dependencies on other jobs and must be submitted in a specified sequence.

Optimizing Unload and Load Job Performance

Upgrades: All upgrades.

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

This topic describes ways in which you can improve the performance of the unload and load jobs for the production upgrade.

■ Optimize the unload and load jobs to reach maximum parallelism:

- Run as many of the unload / load jobs in parallel as the DB2 subsystem can support.
- Change the generated REXX exec job submission order to submit the longest running unload/load jobs first.

If all the unload jobs are run in parallel, the shortest amount of time this process can take is the length of time it takes for the longest unload job to complete.

- For partitioned tables, split the unload files so that data is unloaded and loaded in parallel for each partition. Add the WHERE clause to the unload SQL to control the data that is unloaded.
- Overlap load and unload jobs.

Once an unload job for a table has completed, the load job for that table can be started (assuming you have a one-table per table space database schema structure). This means that load jobs can be running at the same time as unload jobs.

■ Add the ORDER BY clauses to the unload SQL to load data in clustering sequence (you must manually add ORDER BY clauses to the unload SQL).

The *.pretedit.jcl (pretkeys) job builds ORDER BY clauses for individual tables into the data set *.syskeys.orderby.

■ Use third-party utilities to accelerate the unload/load process.

NOTE: You are required to contact Siebel Expert Services before using third-party utilities.

You can use the DB2 Cross Loader (an option of the IBM Load utility) to load data directly from the source to the target database, thereby eliminating the unload step.

■ Populate new columns as part of the unload SQL.

NOTE: You are required to contact Siebel Expert Services before doing this.

■ Add any large tables to *.TABLIST so that the unload and load processes use the large proc, SIEBEL.PROC(SV7LD10L), which allocates more memory, instead of using the standard proc, SIEBEL.PROC(SV7LD10S).

Adding the Statistics Clause to Load Cards

Upgrades: All upgrades.

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

If your database schema structure follows the 1:1:1 model, and if LOAD REPLACE is specified on a load card (so tables are loaded from scratch), you can improve upgrade performance by collecting statistics while running the load job rather than having to run a separate RUNSTATS job. You can do this by adding the STATISTICS clause to the load cards, for example:

```
STATISTICS TABLE(ALL) INDEX(ALL)
UPDATE ACCESSPATH
```

NOTE: If LOAD RESUME is specified on a load card, you cannot collect statistics while running the load job.

About Deactivating Jobs That Do Not Process Data

Upgrades: All upgrades.

Environments: Development (mainframe-centric) environment and Production environment.

Job Stream Optimization is an optional step in the upgrade process that allows you to optimize the Unload, Load and Index Rebuild processes to ensure ease of job scheduling and to minimize the number of jobs run. Job-Stream Optimization allows you to eliminate Unload, Load and Index Rebuild jobs for empty tables (0 rows).

Unload/Load jobs can only be removed if your database schema uses a single table per table space database schema structure; index rebuild jobs can be removed whether you use one table or multiple tables per table space.

NOTE: Take a backup of the appropriate data sets before deleting jobs that act on tables with no data.

The Job Stream Optimization Utility is panel driven in Siebel v7.8. Select option 4: OPTIMIZATION - JOB-STREAM OPTIMIZATION (OPTIONAL) from the Siebel Upgrade Main Menu). The Job-Stream Optimization menu appears displaying options that allow you to:

- 1 Identify tables with 0 rows
- 2 Identify jobs that can be removed
- 3 Remove jobs

See “Optimizing the Target Job Stream, Part 1” on page 224 and “Optimizing the Target Job Stream, Part 2” on page 242 for information on using the Job-Stream Optimization menu options.

CAUTION: You are required to contact Siebel Expert Services for approval before removing jobs for empty tables. If you do not, you may invalidate your support agreement.

A

Tables Modified or Seeded During Upgrade

This appendix describes the important schema changes in Release 7.8. It contains the following topic:

- [“Important Schema Changes at Release 7.8”](#)

Important Schema Changes at Release 7.8

Upgrades: Releases 7.0.x, 7.5.x, 7.7.x.

In Release 7.8.x, the upgrade includes the following important schema changes.

Customer and Order Management Applications

The following schema changes affect customer and order management applications. This includes Product Administration, Order Management, Quotes, Pricer, and Product Configurator.

New Party Payment Profile Table

The new Party Payment Profile table replaces the old Contact Payment Profile table. The new schema supports account and contact payment profiles. The upgrade migrates the data from the old Contact Payment Profile table to the new table.

Quote Teams

Quotes now support multiple sales representatives, or other positions, associated to a single quote. The existing reference to the sales representative for the quote is migrated to the new intersection table and reused as a primary child column.

Multiple Price Types

Products now support multiple price types (Siebel Business applications only). The new Price Type Code column for existing products is set to One-Time. The new One-Time Charges Sub-Total column in Quote, Order, and Agreement line items is calculated and stored. The new Recurring Charges Sub-Total column in Quote, Order, and Agreement line items is set to 0.

Revision to the Line Item Discount Amount Field

The upgrade process changes how line item discounts are stored. If the data source for a discount amount is Pricer, the upgrade moves the value stored in the Discount Amount Field to the Pricing Adjustment Field.

If the Keep Discount flag is checked, the upgrade process does not make this change. Instead, all items that have this flag checked are treated as manual discounts.

The upgrade process determines the correct Header Discount Amount when both of the following are true:

- Before the upgrade, the line item did not have a Manual Discount Amount, Manual Discount %, or Manual Price Override specified
- The line item has a Header Discount % specified

Attribute Pricing Matrices in Pricer

Attribute pricing matrices in Siebel Pricer are replaced by attribute adjustments. After the database upgrade is complete, users must run a business service method to convert attribute pricing data to attribute adjustments. Obsolete tables and corresponding new tables are shown in [Table 43](#).

Table 43. Attribute Adjustment Tables

Obsolete Table	New Table
S_PRI_MTRX	S_DYN_MTRX
S_PRI_MTRX_ATTR	S_DYN_MTRX_DIM
S_PRI_MTRX_ITEM	S_DYN_MTRX_RL
S_PRI_MTRX_VAL	S_DYN_MTRX_DOM

Net Price, Currency Code, Exchange Date

The calculated Net Price field in Quote, Order, and Agreement line items is replaced by a new column. The upgrade calculates the value based on the basic configuration in the prior release. Quote, Order, and Agreement line items now support currency code and exchange date. The relevant data is migrated from the associated header record.

Effective Dates on Price List Line Items and CP Adjustments

Price List line items and Price List CP adjustments now support effective dates in the user key. The effective start and end date are copied from the associated header record where applicable. If this data is not available, the start date is set to 01-01-1980 and the end date is left as NULL.

Volume Discounts

Volume discounts now support discount amounts as well as the existing method of discount percentage. The new price adjustment type code column is used to specify the type. The existing discount amount column stores the associated value. Both columns are updated to support the existing data.

Bundle Discounts

Bundle factors in pricing models are converted to bundle sequences and discounts in Release 7.8. Obsolete tables and corresponding new tables are shown in [Table 44 on page 397](#).

References to pricing models in price lists and price list items are converted to refer to bundle sequences. All other pricing models must be manually reimplemented as PPS Procedures after the database upgrade.

Table 44. Bundle Discount Tables

Obsolete Tables	New Tables
S_PRI MDL	S_BUNDLE_SEQ
S_PRI MDL_FCTR	<ul style="list-style-type: none"> ■ S_BDL_SEQ_I TEM ■ S_BUNDLE_DI SCNT
S_PRI FCTR_I TM	S_BDL_DI SC_I TEM
S_PRI MDLFCTRVAL	NA
S_PRI MDL_OBJ	NA
S_PRI MDL_OBJCRT	NA
S_PRI FCTI TM_ATR	NA

Product Configurator

Release 7.8 introduces a new infrastructure for Product Configurator. The primary new table for configurator is S_VOD. This table stores the header information for products, classes, and attributes.

Other important changes are as follows:

- S_VOD_VER replaces S_PROD_CFGVER and stores the version information for product, class and attribute objects.
- S_I SS_OBJ_DEF stores the product and class definitions.
- S_I SS_ATTR_DEF stores the global (abstract) attribute definitions.
- S_I SS_ATTR_VAL replaces the concept of S_XA_ATTR. VLDTN_LOV_TYPE_CD and stores the enumerated values for the global attribute.
- S_I SS_OBJ_ATTR stores the relationship between local attributes and both classes and products. This replaces the relationships previously stored in S_XA_ATTR. CLASS_ID and S_PROD_I NT_XA.

Obsolete tables and corresponding new tables are shown in [Table 45](#).

Table 45. Product Configurator Tables

Obsolete Tables	New Tables
S_XA_ATTR	<ul style="list-style-type: none"> ■ S_VOD ■ S_I SS_ATTR_DEF ■ S_I SS_ATTR_VAL ■ S_I SS_OBJ_ATTR
S_XA_ATTR_LANG	S_I SS_OBJ_ATTR_LANG
S_XA_CLASS	S_VOD, S_I SS_OBJ_DEF
S_XA_CLASS_LANG	S_VOD_LANG
S_PROD_CFGVER	<ul style="list-style-type: none"> ■ S_VOD_VER ■ S_I SS_UI_OPTION
S_PROD_INT_XA	S_I SS_OBJ_ATTR
S_PROD_ITEM	S_I SS_SUB_OBJ
S_PROD_ITEM_LANG	S_I SS_SOBJ_LANG
S_CFG_VAR_DEF	S_I SS_OBJ_ITEM
S_CFG_PROP_DEF	S_I SS_OBJ_RSRC
S_CFG_SCRIPT_DEF	S_I SS_OBJ_SCRIPT
S_CFG_RULE_DEF	S_I SS_OBJ_CFGRL
S_CFGRLDEF_LANG	S_I SS_CFRL_LANG
S_CFG_RULENODE	S_I SS_CFRL_NODE
S_CFG_UIGROUP	S_I SS_UIOPT_GRP
S_CFGUIGRP_LANG	S_I SS_UIGR_LANG
S_CFG_UIGRP_ITEM	S_I SS_UIGR_ITEM
S_PROD_CFG_PROP	S_I SS_OBJUI_PROP
S_PRDCFGPR_LANG	S_I SS_UIPR_LANG

Like the obsolete tables, all of the new tables have versions. The upgrade migrates data from the obsolete tables to the new ones and creates new records for objects that were previously not versioned (particular classes, attributes, and products).

During the upgrade, additional data migration occurs as follows:

- S_PROD_INT to S_VOD and S_I SS_OBJ_DEF
- S_PROD_INT_LANG to S_VOD_LANG

S_PROD_INT and S_PROD_INT_LANG continue to be used in Release 7.8. Foreign key references to products still point to S_PROD_INT. The S_PROD_INT.CFG_MODEL_ID column is reused to refer to the associated S_VOD record in the Product Configurator infrastructure.

All the S_*_XA tables, for example S_QUOTE_ITEM_XA, now use the ATTR_NAME column to refer to the attribute associated to the object, for example Quote Item.

ATTR_ID is no longer used. In previous releases, it was used as a reference to S_XA_ATTR, which is obsolete in 7.8.

Captive Finance

Before Release 7.8, a contact was associated with only one bankruptcy. In Release 7.8, the relationship between contact and bankruptcy has been changed to M:M. The model also allows capturing bankruptcies for companies. The upgrade migrates data in two steps:

- Migrates bankruptcy information from S_CONTACT_FNX to S_BANKRUPTCY and S_BK_PARTY.
- Migrates associated attorney information for bankruptcies from S_PARTY_REL to S_BK_PARTY. In S_BK_PARTY, attorney type is Bank Attorney, Trustee Attorney, Debtor Attorney or Other Attorney.

The tables S_CONTACT_FNX and S_PARTY_REL continue to be used for other purposes.

eTraining

The upgrade makes the following schema changes:

- Release 7.8 introduces an automatic wait-list feature. Since existing wait-list records were created manually, the upgrade updates the S_SRC_EVT.AUTO_WAITLIST_FLG to N for these records.
- Moves Max Waitlist Num from S_PROD_INT_CRSE to S_SRC_EVT. This moves wait-list support from the course level to the class level.
- Moves the Allow Waitlist Flag from S_PROD_INT_CRSE to S_SRC_EVT. This moves wait-list support from the course level to the class level.

For existing class records, Max Waitlist Num and Allow Waitlist Flag are set to the values that were present for the corresponding course.

Field Service

Quote, Order, and Agreement line items now support multiple covered assets associated to a single line item. The existing reference to the covered asset for the line item is migrated to an intersection table and reused as a primary child column.

The upgrade creates the following new intersection tables:

- S_AGREE_ITEM_REL from S_AGREE_ITEM
- S_ORDER_ITEM_REL from S_ORDER_ITEM
- S_QUOTE_ITEM_REL from S_QUOTE_ITEM

The S_AGREE_ITEM, S_ORDER_ITEM and S_QUOTE_ITEM tables continue to be used.

Consumer Goods Advanced Planning

Release 7.8 introduces an Account Promotion Category in promotion planning. The new account promotion hierarchy is as follows:

- Account Plan
- Account Promotion
- Account Promotion Category
- Account Promotion Product
- Account Promotion Product Baseline/Shipment

To support this, the upgrade inserts account promotion records in S_SRC. The identifier for each record is SUB_TYPE = PLAN_ACCT_PROMOTION_CATEGORY.

The upgrade makes the following changes:

- For every Account Promotion record in S_SRC, the upgrade inserts a record for Account Promotion Category. The parent of the new record is the Account Promotion record.
- Populates S_SRC_CHNL, an extension table for S_SRC, for Account Promotion Category. S_SRC_CHNL.PAR_ROW_ID points to the new record created in S_SRC. S_SRC_CHNL stores certain attributes for Promotion, Promoted Products, and so on.
- Reparents Account Promotion Product records by pointing S_SRC.PAR_SRC_ID to Account Promotion Category.
- Revises Deals records stored in S_MDF_ALLOC that point to Account Promotion records in S_SRC to point also to Account Promotion Category records.

B

Siebel Marketing Upgrade Reference

This appendix describes the Siebel Marketing data that was obsolete as of Release 7.7. If you are upgrading from Release 7.5.x or earlier, review this information.

This appendix contains the following topics:

- [“Obsolete Siebel Marketing Data”](#)
- [“Obsolete Business Objects in Siebel Marketing” on page 407](#)
- [“Obsolete or Replaced Views in Siebel Marketing” on page 408](#)

Obsolete Siebel Marketing Data

Upgrades: Releases 7.0.x, 7.5.x.

[Table 46](#) lists marketing data that is obsolete as of Release 7.7. In the table, the Comments field explains how the data is handled.

Table 46. Obsolete Siebel Marketing Data as of Release 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 46. Obsolete Siebel Marketing Data as of Release 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 46. Obsolete Siebel Marketing Data as of Release 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_QRY	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 46. Obsolete Siebel Marketing Data as of Release 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 46. Obsolete Siebel Marketing Data as of Release 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 46. Obsolete Siebel Marketing Data as of Release 7.7

Data	Effect of Upgrade	Obsolete Tables	Obsolete Business Components	Comments
Attributes and Attribute Families	Obsolete	S_DD_ATRFAMLVMB S_DD_ATTRFAM S_DD_ATTRFAMLVL S_DD_ATTRFAMVAL S_DD_FIELD S_DD_HIERATR S_DD_HIERATRKEY S_DD_HIERATRLVL S_DD_HIERATRVAL	Attribute Families Attribute Family Resync VBC Attribute Family Value Attribute Filters Detail Attribute Levels Attribute Levels.Sequence (Sequence) Hierarchical Attribute Value Hierarchical Attributes Hierarchy (Fields) Hierarchy (Tree) Hierarchy Levels (Tree) Hierarchy Levels Search (Tree) Hierarchy Search (Fields) Hierarchy Search (Tree) Hierarchy Values (Fields) Hierarchy Values (Tree) Hierarchy.Sequence (Sequence)	Segment criteria are now created using presentation columns ("fields") in the Segment Designer.

Table 46. Obsolete Siebel Marketing Data as of Release 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_PPRPERFMEAS	Snapshot Elements	These tables were used by obsolete Siebel Server Components (Data Dictionary Manager and Marketing Server).

Obsolete Business Objects in Siebel Marketing

Upgrades: Releases 7.0.x, 7.5.x.

Table 47 lists the Siebel Repository business objects that are obsolete as of Release 7.7.

Table 47. Obsolete Siebel Marketing Business Objects as of Release 7.7

Business Object Name	Tools Project
Customer Hierarchies	Server (DD)
DBM Campaign	eMarketing - Campaign Views
DBM Campaign Btn	eMarketing - Campaign Views

Table 47. Obsolete Siebel Marketing Business Objects as of Release 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: Releases 7.0.x, 7.5.x.

Table 48 lists the Release 7.x Siebel Repository views that are obsolete or have been replaced as of Release 7.7.

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 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	
Bound Measures View	Marketing Administration	Bound Measures	Obsolete	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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
Campaign Groups View - Admin (SCW)	Campaign Screen (SCW)	Groups	Inactivated	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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
Campaign Plan Activities View	Campaign Management Screen	Activities	Replaced	Campaign Activity View Admin

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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	
Custom Measure Restriction View	Marketing Administration	Custom Measure > Restriction	Obsolete	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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	
Enewsletter Offer Sections List View	Offer Screen	eNewsletter > Sections	Obsolete	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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	
Joins View	Marketing Administration	Joins	Obsolete	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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	
Program Snapshots View	Program Stages Screen	Snapshot Elements	Obsolete	
Program Stages View	Program Screen	Stages	Inactivated	

Table 48. Release 7.x: Obsolete or Replaced Siebel Marketing Views

Obsolete View Name	Release 7.x Screen	Release 7.x View Caption	Outcome	Replacement View
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	

C

Upgrade Planning Worksheet

This appendix contains the Upgrade Planning Worksheet. Before you upgrade your Siebel application, photocopy this worksheet, complete it, and give a copy to each member of the upgrade team.

The Upgrade Planning Worksheet contains the following sections:

- “Team Lead Summary”
- “DB2 Connect Information”
- “Siebel Development Environment Information” on page 418
- “Siebel Production Environment Information” on page 419
- “z/OS Host System Variables Information (Mainframe Upgrade)” on page 420

Team Lead Summary

Deployment Team Lead:	
Siebel Administrator:	
Privileged User/Siebel Database User:	
DB2 Systems Programmer (SYSADM):	
DB2 Database Administrator (DBADM):	
Security Administrator:	
z/OS System Programmer:	
Midtier System Administrator:	

DB2 Connect Information

DB2 Host Name/IP Address:

■ DB2 Port Number:

Siebel Development Environment Information

Siebel Gateway Name
Server Name: _____

Enterprise Server Name: _____

Siebel Server Directory: _____

Siebel Database Server
Directory _____

Database Alias: _____

Siebel Administrator User Name	Siebel Administrator Password	Siebel Administrator User Group
-----------------------------------	----------------------------------	------------------------------------

Source Siebel Schema Qualifier ID (Max. 8 chars)	Source ODBC Data Source Name (Subsystem name)	Source Database User Name
--	--	------------------------------

Target Siebel Schema Qualifier ID (Max. 8 chars)	Target ODBC Data Source Name (Subsystem name)	Target Database User Name
--	--	------------------------------

Target Security Group ID	EIM User Group ID (Max. 8 characters)	Siebel User Group ID (Max. 8 characters)
--------------------------	--	---

Storage Control File Name	Storage Group for Temporary Indexes	Database Name Prefix (Max. 4 characters)
---------------------------	--	---

4-KB Bufferpool	16-KB Bufferpool	32-KB Bufferpool
-----------------	------------------	------------------

Index Bufferpool _____

NOTE: The Security Group ID is also known as the secondary authorization ID.

Siebel Production Environment Information

Siebel Gateway Name

Server Name: _____

Enterprise Server Name: _____

Siebel Server Directory: _____

Siebel Database Server

Directory: _____

Database Alias: _____

Siebel Administrator User
Name

Siebel Administrator
Password

Siebel Administrator User
Group

Source Siebel Schema
Qualifier ID (Max. 8 chars)

Source ODBC Data Source
Name (Subsystem name)

Source Database User
Name

Target Siebel Schema
Qualifier ID (Max. 8 chars)

Target ODBC Data Source
Name (Subsystem name)

Target Database User
Name

Target Security Group ID

EIM User Group ID (Max. 8
characters)

Siebel User Group ID
(Max. 8 characters)

Storage Control File Name

Storage Group for
Temporary Indexes

Database Name Prefix
(Max. 4 characters)

ODBC DSN for Development
Database

Database User Name for
Development Database

Database Table Owner for
Development Database

Import Repository Name

4-KB Bufferpool

16-KB Bufferpool

32-KB Bufferpool

Index Bufferpool

NOTE: The Security Group ID is also known as the secondary authorization ID.

z/OS Host System Variables Information (Mainframe Upgrade)

**DSN High Level Qualifier
Name (DSNHLO)**

Host/LPAR Name:

DB2 WLM Name:

Code Page / CCSID

**DB2 Load Libraries:
Source Database**

**DB2 Load Libraries:
Target Database**

To obtain the correct values for the system variables, talk to your DBA or systems programmer.

D

Columns Denormalized During Upgrade to Release 7.8

This appendix lists columns that are denormalized during upgrades to Release 7.8 Siebel Industry applications. This appendix contains the following topics:

- [“Denormalized Columns for 6.2.1 Siebel Financial Services Applications”](#)
- [“Denormalized Columns for 6.2.1 Siebel eBusiness Applications” on page 430](#)
- [“Denormalized Columns for 6.3 Siebel Industry Solutions” on page 438](#)
- [“Denormalized Columns for 7.0.4 Siebel eBusiness Applications” on page 448](#)
- [“Denormalized Columns for 7.0.4 Siebel Financial Services Applications” on page 450](#)
- [“Denormalized Columns for 7.0.4 Siebel Industry Solutions” on page 454](#)
- [“Denormalized Columns for 7.5.2 Siebel Industry Applications” on page 459](#)

NOTE: If you reduced column lengths when you installed the Siebel Database Server on DB2 UDB for z/OS, you must review them before upgrading to Release 7.8. The upgrade does not recognize the denormalized columns.

Denormalized Columns for 6.2.1 Siebel Financial Services Applications

Table 49 lists columns that are denormalized during upgrades from Release 6.2.1 Siebel Financial Services applications to Release 7.8 Siebel Industry applications.

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ACCNT_TYPE_CD	[OU_EXT_ID].[ACCNT_TYPE_CD]	S_ORG_EXT	ACCNT_TYPE_CD
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_CAL_RSRC	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_CAL_RSRC	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACT_CAL_RSRC	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT
S_ACT_CAL_RSRC	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_ACT_EMP	ACT_ALARM_FLG	[ACTIVITY_ID].[ALARM_FLAG]	S_EVT_ACT	ALARM_FLAG
S_ACT_EMP	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_EMP	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_EMP	ACT_APPT_START_DT	[ACTIVITY_ID].[APPT_START_DT]	S_EVT_ACT	APPT_START_DT
S_ACT_EMP	ACT_CAL_DISP_FLG	[ACTIVITY_ID].[CAL_DISP_FLG]	S_EVT_ACT	CAL_DISP_FLG
S_ACT_EMP	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_EMP	ACT_EVT_STAT_CD	[ACTIVITY_ID].[EVT_STAT_CD]	S_EVT_ACT	EVT_STAT_CD
S_ACT_EMP	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_EMP	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT
S_ACT_EMP	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_AGREE_POSTN	AGREE_NAME	[AGREE_ID].[NAME]	S_DOC_AGREE	NAME
S_AGREE_POSTN	AGREE_STAT_CD	[AGREE_ID].[STAT_CD]	S_DOC_AGREE	STAT_CD
S_AGREE_POSTN	AGREE_VALID_FLG	[AGREE_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_ASSET_BU	ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_ASSET_BU	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_CASE_BU	CASE_NAME	[CASE_ID].[NAME]	S_CASE	NAME
S_CASE_BU	CASE_STATUS_CD	[CASE_ID].[STATUS_CD]	S_CASE	STATUS_CD
S_CASE_BU	CASE_TYPE_CD	[CASE_ID].[TYPE_CD]	S_CASE	TYPE_CD
S_CONTACT_BU	AGENT_FLG	[CONTACT_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_CONTACT_BU	CON_FST_NAME	[CONTACT_ID].[FST_NAME]	S_CONTACT	FST_NAME
S_CONTACT_BU	CON_LAST_NAME	[CONTACT_ID].[LAST_NAME]	S_CONTACT	LAST_NAME
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	MEMBER_FLG	[CONTACT_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_CONTACT_BU	PROVIDER_FLG	[CONTACT_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_CTLG_CAT_AUC	AUC_AUC_LOT_NUM	[AUC_ITEM_ID].[AUC_LOT_NUM]	S_AUC_ITEM	AUC_LOT_NUM
S_CTLG_CAT_CRSE	CRSE_NAME	[CRSE_ID].[NAME]	S_CRSE	NAME
S_CTLG_CAT_CRSE	CRSE_SUB_TYPE	[CRSE_ID].[SUB_TYPE]	S_CRSE	SUB_TYPE
S_CTLG_CAT_DFCT	DFCT_DFCT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM
S_CTLG_CAT_ISS	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_CTLG_CAT_LIT	LIT_NAME	[LIT_ID].[NAME]	S_LIT	NAME
S_CTLG_CAT_ORG	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_CTLG_CAT_ORG	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_CTLG_CAT_ORG	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME
S_CTLG_CAT_ORG	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_CTLG_CAT_PROD	PROD_EFF_END_DT	[PROD_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_CTLG_CAT_PROD	PROD_EFF_START_DT	[PROD_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_CTLG_CAT_PROD	PROD_NAME	[PROD_ID].[NAME]	S_PROD_INT	NAME
S_CTLG_CAT_SR	SR_SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_CTLG_CAT_SRC	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_CTLG_CAT_SRC	SRC_SRC_NUM	[SRC_ID].[SRC_NUM]	S_SRC	SRC_NUM
S_CTLG_CAT_SRC	SRC_SUB_TYPE	[SRC_ID].[SUB_TYPE]	S_SRC	SUB_TYPE
S_CTLGCAT_ASSET	ASSET_ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_CTLGCAT_ASSET	ASSET_SERIAL_NUM	[ASSET_ID].[SERIAL_NUM]	S_ASSET	SERIAL_NUM
S_CTLGCAT_PATH	CS_PATH_NAME	[CS_PATH_ID].[NAME]	S_CS_PATH	NAME
S_CTLGCT_RESI_TM	RI_INTR_PUBL_FLG	[RES_ITEM_ID].[INTR_PUBLISH_FLG]	S_RES_ITEM	INTR_PUBLISH_FLG
S_CTLGCT_RESI_TM	RI_NAME	[RES_ITEM_ID].[NAME]	S_RES_ITEM	NAME
S_CTLGCT_RESI_TM	RI_TYPE_CD	[RES_ITEM_ID].[TYPE_CD]	S_RES_ITEM	TYPE_CD
S_DOC_AGREE_BU	AGREE_NAME	[AGREEMENT_ID].[NAME]	S_DOC_AGREE	NAME
S_DOC_AGREE_BU	AGREE_STAT_CD	[AGREEMENT_ID].[STAT_CD]	S_DOC_AGREE	STAT_CD
S_DOC_AGREE_BU	AGREE_VALID_FLG	[AGREEMENT_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_DOC_PPSL_BU	PPSL_DFLT_TMPL_FLG	[DOC_PPSL_ID].[DFLT_TEMPLATE_FLG]	S_DOC_PPSL	DFLT_TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_NAME	[DOC_PPSL_ID].[NAME]	S_DOC_PPSL	NAME

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_DOC_PPSL_BU	PPSL_TEMPLATE_FLG	[DOC_PPSL_ID].[TEMPLATE_FLG]	S_DOC_PPSL	TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_TYPE_CD	[DOC_PPSL_ID].[PPSL_TYPE_CD]	S_DOC_PPSL	PPSL_TYPE_CD
S_DOC_QUOTE_BU	DOC_QTE_NUM	[DOC_QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_DOC_QUOTE_BU	DOC_QTE_REV_NUM	[DOC_QUOTE_ID].[REV_NUM]	S_DOC_QUOTE	REV_NUM
S_DOC_QUOTE_BU	DOC_QUOTE_NAME	[DOC_QUOTE_ID].[NAME]	S_DOC_QUOTE	NAME
S_INSCLM_BU	INSCLAIM_NUM	[INSCLM_ID].[INSCLAIM_NUM]	S_INSCLAIM	INSCLAIM_NUM
S_INSCLM_BU	REV_NUM	[INSCLM_ID].[REVISION_NUM]	S_INSCLAIM	REVISION_NUM
S_INSCLMEL_BU	SEQ_NUM	[INSCLM_ELMNT_ID].[SEQ_NUM]	S_INSCLM_ELMNT	SEQ_NUM
S_INSCLMEL_BU	TYPE_CD	[INSCLM_ELMNT_ID].[TYPE_CD]	S_INSCLM_ELMNT	TYPE_CD
S_INV_TXN_BU	INV_TXN_NUM	[INV_TXN_ID].[INV_TXN_NUM]	S_INV_TXN	INV_TXN_NUM
S_ISS_BU	ISS_LANG_ID	[ISS_ID].[LANG_ID]	S_ISS	LANG_ID
S_ISS_BU	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_LOY_MEM_BU	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_BU	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_MEM_PSTN	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_PSTN	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_PROG_BU	PROG_NAME	[PROG_ID].[NAME]	S_LOY_PROGRAM	NAME
S_LOY_PROMO_BU	PROMO_NAME	[PROMO_ID].[NAME]	S_LOY_PROMO	NAME
S_LOY_PROMO_BU	PROMO_NUM	[PROMO_ID].[PROMO_NUM]	S_LOY_PROMO	PROMO_NUM

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_LOY_TXN_BU	TXN_NUM	[TXN_ID].[TXN_NUM]	S_LOY_TXN	TXN_NUM
S_LOY_TXN_BU	TXN_STATUS_CD	[TXN_ID].[STATUS_CD]	S_LOY_TXN	STATUS_CD
S_LOY_TXN_BU	TXN_SUB_TYPE_CD	[TXN_ID].[SUB_TYPE_CD]	S_LOY_TXN	SUB_TYPE_CD
S_LOY_TXN_BU	TXN_TYPE_CD	[TXN_ID].[TYPE_CD]	S_LOY_TXN	TYPE_CD
S_MDF_BU	MDF_UI D	[MDF_ID].[MDF_UI D]	S_MDF	MDF_UI D
S_OPTY_BU	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME
S_OPTY_BU	SUM_CLASS_CD	[OPTY_ID].[SUM_CLASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_BU	SUM_COMMIT_FLG	[OPTY_ID].[SUM_COMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG
S_OPTY_BU	SUM_COST_AMT	[OPTY_ID].[SUM_COST_AMT]	S_OPTY	SUM_COST_AMT
S_OPTY_BU	SUM_DOWNSIDE_AMT	[OPTY_ID].[SUM_DOWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_AMT
S_OPTY_BU	SUM_EFFECTIVE_DT	[OPTY_ID].[SUM_EFFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_DT
S_OPTY_BU	SUM_MARGIN_AMT	[OPTY_ID].[SUM_MARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_BU	SUM_REVN_AMT	[OPTY_ID].[SUM_REVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_BU	SUM_TYPE_CD	[OPTY_ID].[SUM_TYPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_BU	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_UPSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_BU	SUM_WIN_PROB	[OPTY_ID].[SUM_WIN_PROB]	S_OPTY	SUM_WIN_PROB
S_OPTY_POSTN	NEW_LOAN_FLG	[OPTY_ID].[NEW_LOAN_FLG]	S_OPTY	NEW_LOAN_FLG
S_OPTY_POSTN	OPTY_CLOSED_FLG	[OPTY_ID].[CLOSED_FLG]	S_OPTY	CLOSED_FLG
S_OPTY_POSTN	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_OPTY_POSTN	SUM_CLASS_CD	[OPTY_ID].[SUM_CLASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_POSTN	SUM_COMMIT_FLG	[OPTY_ID].[SUM_COMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG
S_OPTY_POSTN	SUM_COST_AMT	[OPTY_ID].[SUM_COST_AMT]	S_OPTY	SUM_COST_AMT
S_OPTY_POSTN	SUM_DOWNSIDE_AMT	[OPTY_ID].[SUM_DOWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_AMT
S_OPTY_POSTN	SUM_EFFECTIVE_DT	[OPTY_ID].[SUM_EFFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_DT
S_OPTY_POSTN	SUM_MARGIN_AMT	[OPTY_ID].[SUM_MARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_POSTN	SUM_REVN_AMT	[OPTY_ID].[SUM_REVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_POSTN	SUM_TYPE_CD	[OPTY_ID].[SUM_TYPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_POSTN	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_UPSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_POSTN	SUM_WIN_PROB	[OPTY_ID].[SUM_WIN_PROB]	S_OPTY	SUM_WIN_PROB
S_ORDER_BU	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_BU	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_BU	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORDER_POSTN	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_POSTN	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_POSTN	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORG_BU	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_ORG_BU	ORG_FACILITY_FLG	[ORG_ID].[FACILITY_FLG]	S_ORG_EXT	FACILITY_FLG

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ORG_BU	ORG_INVSTR_FLG	[ORG_ID].[INVSTR_FLG]	S_ORG_EXT	INVSTR_FLG
S_ORG_BU	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_ORG_BU	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME
S_ORG_BU	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_BU	ORG_SRV_PROVDR_FLG	[ORG_ID].[SRV_PROVDR_FLG]	S_ORG_EXT	SRV_PROVDR_FLG
S_ORG_DIST_LST	OU_ID	[ORG_PROD_ID].[OU_ID]	S_ORG_PROD	OU_ID
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_ORG_GROUP_BU	OG_GROUP_TYPE_CD	[ORG_GROUP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GROUP_TYPE_CD	[ORGGRP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_PERIOD_BU	PERIOD_NAME	[PERIOD_ID].[NAME]	S_PERIOD	NAME
S_POS_BU	POS_NUM	[POS_ID].[POS_NUM]	S_POS	POS_NUM
S_POSTN_CON	AGENT_FLG	[CON_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_POSTN_CON	MEMBER_FLG	[CON_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_PRIMFCTR_ITM	PRIMDL_ID	[PRIMDL_FCTR_ID].[PRIMDL_ID]	S_PRIMDL_FCTR	PRIMDL_ID
S_PROD_DFCT_BU	DEFECT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_PROD_INT_BU	PROD_EFF_END_DT	[PROD_INT_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_PROD_INT_BU	PROD_EFF_START_DT	[PROD_INT_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_PROD_STYL_TNT	SETUP_STYLE_CD	[PROP_STYLE_ID].[SETUP_STYLE_CD]	S_PROD_STYL_TNT	SETUP_STYLE_CD
S_PROJ_BU	PROJ_NAME	[PROJ_ID].[NAME]	S_PROJ	NAME
S_PROJ_BU	PROJ_STATUS_CD	[PROJ_ID].[STATUS_CD]	S_PROJ	STATUS_CD
S_PROJ_BU	PROJ_TYPE_CD	[PROJ_ID].[PROJ_TYPE_CD]	S_PROJ	PROJ_TYPE_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SALES_MTHD_BU	SALES_METHOD_NAME	[SALES_METHOD_ID].[NAME]	S_SALES_METHOD	NAME
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD
S_TMPL_PLNIT_BU	PLANITEM_NAME	[TMPL_PLANITEM_ID].[NAME]	S_TMPL_PLNITEM	NAME

Table 49. Columns Denormalized During Upgrades from Release 6.2.1 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_TMPL_PLNIT_BU	PLANITEM_TYPE_CD	[TMPL_PLANITEM_ID].[TYPE_CD]	S_TMPL_PLNITEM	TYPE_CD
S_TMPL_PLNIT_BU	PLNIT_TMPL_TYPE_CD	[TMPL_PLANITEM_ID].[TMPL_TYPE_CD]	S_TMPL_PLNITEM	TMPL_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
S_WRNTY_CVRG_BU	WRNTY_CVRG_NAME	[WRNTY_CVRG_ID].[WRNTY_NAME]	S_WRNTY_CVRG	WRNTY_NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 6.2.1 Siebel eBusiness Applications

Table 50 lists columns that are denormalized during upgrades from Release 6.2.1 Siebel eBusiness applications to Release 7.8 Siebel Industry applications.

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ORG_REF_CUST_FLG	[OU_EXT_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACT_CAL_RSRC	ACT_APPT_RPTEN D_DT	[ACTIVITY_ID].[AP PT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_D T
S_ACT_CAL_RSRC	ACT_CAL_TYPE_C D	[ACTIVITY_ID].[CA L_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_F LG	[ACTIVITY_ID].[TE MPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_CAL_RSRC	ACT_TODO_PLNEN D_DT	[ACTIVITY_ID].[TO DO_PLAN_END_DT]	S_EVT_ACT	TODD_PLAN_END_D T
S_ACT_CAL_RSRC	ACT_TODO_PLNST RTDT	[ACTIVITY_ID].[TO DO_PLAN_START_D T]	S_EVT_ACT	TODD_PLAN_START _DT
S_ACT_EMP	ACT_ALARM_FLG	[ACTIVITY_ID].[AL ARM_FLAG]	S_EVT_ACT	ALARM_FLAG
S_ACT_EMP	ACT_APPT_REPT_ FLG	[ACTIVITY_ID].[AP PT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_EMP	ACT_APPT_RPTEN D_DT	[ACTIVITY_ID].[AP PT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_D T
S_ACT_EMP	ACT_APPT_START _DT	[ACTIVITY_ID].[AP PT_START_DT]	S_EVT_ACT	APPT_START_DT
S_ACT_EMP	ACT_CAL_DISP_F LG	[ACTIVITY_ID].[CA L_DISP_FLG]	S_EVT_ACT	CAL_DISP_FLG
S_ACT_EMP	ACT_EVT_STAT_C D	[ACTIVITY_ID].[EV T_STAT_CD]	S_EVT_ACT	EVT_STAT_CD
S_ACT_EMP	ACT_TEMPLATE_F LG	[ACTIVITY_ID].[TE MPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_EMP	ACT_TODO_PLNEN D_DT	[ACTIVITY_ID].[TO DO_PLAN_END_DT]	S_EVT_ACT	TODD_PLAN_END_D T
S_ACT_EMP	ACT_TODO_PLNST RTDT	[ACTIVITY_ID].[TO DO_PLAN_START_D T]	S_EVT_ACT	TODD_PLAN_START _DT
S_ACT_EMP	ACT_CAL_TYPE_C D	[ACTIVITY_ID].[CA L_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_AGREE_POSTN	AGREE_NAME	[AGREE_ID].[NAME]	S_DOC_AGREE	NAME
S_AGREE_POSTN	AGREE_STAT_CD	[AGREE_ID].[STAT _CD]	S_DOC_AGREE	STAT_CD

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_AGREE_POSTN	AGREE_VALID_FLG	[AGREE_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_CONTACT_BU	CON_FST_NAME	[CONTACT_ID].[FST_NAME]	S_CONTACT	FST_NAME
S_CONTACT_BU	CON_LAST_NAME	[CONTACT_ID].[LAST_NAME]	S_CONTACT	LAST_NAME
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_CTLG_CAT_AUC	AUC_AUC_LOT_NUM	[AUC_ITEM_ID].[AUC_LOT_NUM]	S_AUC_ITEM	AUC_LOT_NUM
S_CTLG_CAT_CRSE	CRSE_NAME	[CRSE_ID].[NAME]	S_CRSE	NAME
S_CTLG_CAT_CRSE	CRSE_SUB_TYPE	[CRSE_ID].[SUB_TYPE]	S_CRSE	SUB_TYPE
S_CTLG_CAT_DFCT	DFCT_DFCT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM
S_CTLG_CAT_ISS	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_CTLG_CAT_LIT	LIT_NAME	[LIT_ID].[NAME]	S_LIT	NAME
S_CTLG_CAT_ORG	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_CTLG_CAT_ORG	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_CTLG_CAT_ORG	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME
S_CTLG_CAT_ORG	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG
S_CTLG_CAT_PROD	PROD_EFF_END_DT	[PROD_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_CTLG_CAT_PROD	PROD_EFF_START_DT	[PROD_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_CTLG_CAT_PROD	PROD_NAME	[PROD_ID].[NAME]	S_PROD_INT	NAME
S_CTLG_CAT_SR	SR_SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_CTLG_CAT_SRC	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_CTLG_CAT_SRC	SRC_SRC_NUM	[SRC_ID].[SRC_NUM]	S_SRC	SRC_NUM

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_CTLG_CAT_SRC	SRC_SUB_TYPE	[SRC_ID].[SUB_TYPE]	S_SRC	SUB_TYPE
S_CTLGCAT_ASSET	ASSET_ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_CTLGCAT_ASSET	ASSET_SERIAL_NUM	[ASSET_ID].[SERIAL_NUM]	S_ASSET	SERIAL_NUM
S_CTLGCAT_PATH	CS_PATH_NAME	[CS_PATH_ID].[NAME]	S_CS_PATH	NAME
S_CTLGCT_RESITEM	RI_INTR_PUBL_FLG	[RES_ITEM_ID].[INTR_PUBLISH_FLG]	S_RESITEM	INTR_PUBLISH_FLG
S_CTLGCT_RESITEM	RI_NAME	[RES_ITEM_ID].[NAME]	S_RESITEM	NAME
S_CTLGCT_RESITEM	RI_TYPE_CD	[RES_ITEM_ID].[TYPE_CD]	S_RESITEM	TYPE_CD
S_DOC_AGREE_BU	AGREE_NAME	[AGREEMENT_ID].[NAME]	S_DOC_AGREE	NAME
S_DOC_AGREE_BU	AGREE_STAT_CD	[AGREEMENT_ID].[STAT_CD]	S_DOC_AGREE	STAT_CD
S_DOC_AGREE_BU	AGREE_VALID_FLG	[AGREEMENT_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_DOC_PPSL_BU	PPSL_DFLT_TMPL_FLG	[DOC_PPSL_ID].[DFLT_TEMPLATE_FLG]	S_DOC_PPSL	DFLT_TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_NAME	[DOC_PPSL_ID].[NAME]	S_DOC_PPSL	NAME
S_DOC_PPSL_BU	PPSL_TEMPLATE_FLG	[DOC_PPSL_ID].[TEMPLATE_FLG]	S_DOC_PPSL	TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_TYPE_CD	[DOC_PPSL_ID].[PPSL_TYPE_CD]	S_DOC_PPSL	PPSL_TYPE_CD
S_DOC_QUOTE_BU	DOC_QTE_NUM	[DOC_QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_DOC_QUOTE_BU	DOC_QTE_REV_NUM	[DOC_QUOTE_ID].[REV_NUM]	S_DOC_QUOTE	REV_NUM
S_DOC_QUOTE_BU	DOC_QUOTE_NAME	[DOC_QUOTE_ID].[NAME]	S_DOC_QUOTE	NAME

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_INV_TXN_BU	INV_TXN_NUM	[INV_TXN_ID].[INV_TXN_NUM]	S_INV_TXN	INV_TXN_NUM
S_ISS_BU	ISS_LANG_ID	[ISS_ID].[LANG_ID]	S_ISS	LANG_ID
S_ISS_BU	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_MDF_BU	MDF_UI D	[MDF_ID].[MDF_UI D]	S_MDF	MDF_UI D
S_OPTY_BU	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME
S_OPTY_BU	SUM_CLASS_CD	[OPTY_ID].[SUM_C LASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_BU	SUM_COMMIT_FLG	[OPTY_ID].[SUM_C OMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG
S_OPTY_BU	SUM_COST_AMT	[OPTY_ID].[SUM_C OST_AMT]	S_OPTY	SUM_COST_AMT
S_OPTY_BU	SUM_DOWNSIDE_A MT	[OPTY_ID].[SUM_D OWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_A MT
S_OPTY_BU	SUM_EFFECTIVE_D T	[OPTY_ID].[SUM_E FFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_D T
S_OPTY_BU	SUM_MARGIN_AMT	[OPTY_ID].[SUM_M ARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_BU	SUM_REVN_AMT	[OPTY_ID].[SUM_R EVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_BU	SUM_TYPE_CD	[OPTY_ID].[SUM_T YPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_BU	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_U PSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_BU	SUM_WIN_PROB	[OPTY_ID].[SUM_W IN_PROB]	S_OPTY	SUM_WIN_PROB
S_OPTY_POSTN	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME
S_OPTY_POSTN	SUM_CLASS_CD	[OPTY_ID].[SUM_C LASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_POSTN	SUM_COMMIT_FLG	[OPTY_ID].[SUM_C OMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG
S_OPTY_POSTN	SUM_COST_AMT	[OPTY_ID].[SUM_C OST_AMT]	S_OPTY	SUM_COST_AMT

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_OPTY_POSTN	SUM_DOWNSIDE_AMT	[OPTY_ID].[SUM_DOWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_AMT
S_OPTY_POSTN	SUM_EFFECTIVE_DT	[OPTY_ID].[SUM_EFFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_DT
S_OPTY_POSTN	SUM_MARGIN_AMT	[OPTY_ID].[SUM_MARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_POSTN	SUM_REVN_AMT	[OPTY_ID].[SUM_REVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_POSTN	SUM_TYPE_CD	[OPTY_ID].[SUM_TYPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_POSTN	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_UPSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_POSTN	SUM_WIN_PROB	[OPTY_ID].[SUM_WIN_PROB]	S_OPTY	SUM_WIN_PROB
S_ORDER_BU	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_BU	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_BU	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORDER_POSTN	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_POSTN	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_POSTN	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORG_BU	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_ORG_BU	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_ORG_BU	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME
S_ORG_BU	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GRP_NAME	[ORGGRP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_PERIOD_BU	PERIOD_NAME	[PERIOD_ID].[NAME]	S_PERIOD	NAME
S_PRIMFCTR_ITM	PRIMDL_ID	[PRIMDL_FCTR_ID].[PRIMDL_ID]	S_PRIMDL_FCTR	PRIMDL_ID
S_PROD_DFCT_BU	DEFECT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM
S_PROD_INT_BU	PROD_EFF_END_DT	[PROD_INT_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_PROD_INT_BU	PROD_EFF_START_DT	[PROD_INT_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_PROJ_BU	PROJ_NAME	[PROJ_ID].[NAME]	S_PROJ	NAME
S_PROJ_BU	PROJ_TYPE_CD	[PROJ_ID].[PROJ_TYPE_CD]	S_PROJ	PROJ_TYPE_CD
S_PROJ_BU	PROJ_STATUS_CD	[PROJ_ID].[STATUS_CD]	S_PROJ	STATUS_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SALES_MTHD_BU	SALES_METHOD_NAME	[SALES_METHOD_ID].[NAME]	S_SALES_METHOD	NAME
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID

Table 50. Columns Denormalized During Upgrades from Release 6.2.1 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD
S_TMPL_PLNIT_BU	PLANITEM_NAME	[TMPL_PLANITEM_ID].[NAME]	S_TMPL_PLANITEM	NAME
S_TMPL_PLNIT_BU	PLANITEM_TYPE_CD	[TMPL_PLANITEM_ID].[TYPE_CD]	S_TMPL_PLANITEM	TYPE_CD
S_TMPL_PLNIT_BU	PLNIT_TMPL_TYPE_CD	[TMPL_PLANITEM_ID].[TMPL_TYPE_CD]	S_TMPL_PLANITEM	TMPL_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
S_WRNTY_CVRG_BU	WRNTY_CVRG_NAME	[WRNTY_CVRG_ID].[WRNTY_NAME]	S_WRNTY_CVRG	WRNTY_NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 6.3 Siebel Industry Solutions

Table 51 lists columns that are denormalized during upgrades from Release 6.3 Siebel Industry Solutions to Release 7.8 Siebel Industry applications.

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ACCNT_TYPE_CD	[OU_EXT_ID].[ACCNT_TYPE_CD]	S_ORG_EXT	ACCNT_TYPE_CD
S_ACCNT_POSTN	CLIENT_FLG	[OU_EXT_ID].[CLIENT_FLG]	S_ORG_EXT	CLIENT_FLG
S_ACCNT_POSTN	FACILITY_FLG	[OU_EXT_ID].[FACILITY_FLG]	S_ORG_EXT	FACILITY_FLG
S_ACCNT_POSTN	INVSTR_FLG	[OU_EXT_ID].[INVSTR_FLG]	S_ORG_EXT	INVSTR_FLG
S_ACCNT_POSTN	ORG_REF_CUST_FLG	[OU_EXT_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ACCNT_POSTN	SRV_PROVDR_FLG	[OU_EXT_ID].[SRV_PROVDR_FLG]	S_ORG_EXT	SRV_PROVDR_FLG
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_CAL_RSRC	ACT_APPT_RPTENDD_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_CAL_RSRC	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_CAL_RSRC	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODD_PLAN_END_DT
S_ACT_CAL_RSRC	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODD_PLAN_START_DT
S_ACT_EMP	ACT_ALARM_FLG	[ACTIVITY_ID].[ALARM_FLAG]	S_EVT_ACT	ALARM_FLAG
S_ACT_EMP	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACT_EMP	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_EMP	ACT_APPT_START_DT	[ACTIVITY_ID].[APPT_START_DT]	S_EVT_ACT	APPT_START_DT
S_ACT_EMP	ACT_CAL_DISP_FLG	[ACTIVITY_ID].[CAL_DISP_FLG]	S_EVT_ACT	CAL_DISP_FLG
S_ACT_EMP	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_EMP	ACT_EVT_STAT_CD	[ACTIVITY_ID].[EVT_STAT_CD]	S_EVT_ACT	EVT_STAT_CD
S_ACT_EMP	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_EMP	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT
S_ACT_EMP	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_AGREE_POSTN	AGREE_NAME	[AGREE_ID].[NAME]	S_DOC_AGREE	NAME
S_AGREE_POSTN	AGREE_STAT_CD	[AGREE_ID].[STAT_CD]	S_DOC_AGREE	STAT_CD
S_AGREE_POSTN	AGREE_VALID_FLG	[AGREE_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_ASSET_BU	ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_ASSET_BU	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD
S_ASSET_POSTN	ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_ASSET_POSTN	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD
S_CASE_BU	CASE_NAME	[CASE_ID].[NAME]	S_CASE	NAME
S_CASE_BU	CASE_STATUS_CD	[CASE_ID].[STATUS_CD]	S_CASE	STATUS_CD
S_CASE_BU	CASE_TYPE_CD	[CASE_ID].[TYPE_CD]	S_CASE	TYPE_CD

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_CONTACT_BU	AGENT_FLG	[CONTACT_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_CONTACT_BU	CON_FST_NAME	[CONTACT_ID].[FST_NAME]	S_CONTACT	FST_NAME
S_CONTACT_BU	CON_LAST_NAME	[CONTACT_ID].[LAST_NAME]	S_CONTACT	LAST_NAME
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	MEMBER_FLG	[CONTACT_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_CONTACT_BU	PROVIDER_FLG	[CONTACT_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_CTLG_CAT_AUC	AUC_AUC_LOT_NUM	[AUC_ITEM_ID].[AUC_LOT_NUM]	S_AUC_ITEM	AUC_LOT_NUM
S_CTLG_CAT_CRSE	CRSE_NAME	[CRSE_ID].[NAME]	S_CRSE	NAME
S_CTLG_CAT_CRSE	CRSE_SUB_TYPE	[CRSE_ID].[SUB_TYPE]	S_CRSE	SUB_TYPE
S_CTLG_CAT_DFCT	DFCT_DFCT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM
S_CTLG_CAT_ISS	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_CTLG_CAT_LIT	LIT_NAME	[LIT_ID].[NAME]	S_LIT	NAME
S_CTLG_CAT_ORG	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_CTLG_CAT_ORG	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_CTLG_CAT_ORG	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME
S_CTLG_CAT_ORG	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG
S_CTLG_CAT_PROD	PROD_EFF_END_DT	[PROD_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_CTLG_CAT_PROD	PROD_EFF_START_DT	[PROD_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_CTLG_CAT_PROD	PROD_NAME	[PROD_ID].[NAME]	S_PROD_INT	NAME
S_CTLG_CAT_SR	SR_SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_CTLG_CAT_SRC	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_CTLG_CAT_SRC	SRC_SRC_NUM	[SRC_ID].[SRC_NUM]	S_SRC	SRC_NUM
S_CTLG_CAT_SRC	SRC_SUB_TYPE	[SRC_ID].[SUB_TYPE]	S_SRC	SUB_TYPE
S_CTLGCAT_ASSET	ASSET_ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_CTLGCAT_ASSET	ASSET_SERIAL_NUM	[ASSET_ID].[SERIAL_NUM]	S_ASSET	SERIAL_NUM
S_CTLGCAT_PATH	CS_PATH_NAME	[CS_PATH_ID].[NAME]	S_CS_PATH	NAME
S_CTLGCT_RESI_TM	RI_INTR_PUBL_FLG	[RES_ITEM_ID].[INTR_PUBLISH_FLG]	S_RES_ITEM	INTR_PUBLISH_FLG
S_CTLGCT_RESI_TM	RI_NAME	[RES_ITEM_ID].[NAME]	S_RES_ITEM	NAME
S_CTLGCT_RESI_TM	RI_TYPE_CD	[RES_ITEM_ID].[TYPE_CD]	S_RES_ITEM	TYPE_CD
S_DOC_AGREE_BU	AGREE_NAME	[AGREEMENT_ID].[NAME]	S_DOC_AGREE	NAME
S_DOC_AGREE_BU	AGREE_STAT_CD	[AGREEMENT_ID].[STAT_CD]	S_DOC_AGREE	STAT_CD
S_DOC_AGREE_BU	AGREE_VALID_FLG	[AGREEMENT_ID].[VALID_FLG]	S_DOC_AGREE	VALID_FLG
S_DOC_PPSL_BU	PPSL_DFLT_TMPL_FLG	[DOC_PPSL_ID].[DFLT_TEMPLATE_FLG]	S_DOC_PPSL	DFLT_TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_NAME	[DOC_PPSL_ID].[NAME]	S_DOC_PPSL	NAME
S_DOC_PPSL_BU	PPSL_TEMPLATE_FLG	[DOC_PPSL_ID].[TEMPLATE_FLG]	S_DOC_PPSL	TEMPLATE_FLG
S_DOC_PPSL_BU	PPSL_TYPE_CD	[DOC_PPSL_ID].[PPSL_TYPE_CD]	S_DOC_PPSL	PPSL_TYPE_CD
S_DOC_QUOTE_BU	DOC_QTE_NUM	[DOC_QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_DOC_QUOTE_BU	DOC_QTE_REV_NUM	[DOC_QUOTE_ID].[REV_NUM]	S_DOC_QUOTE	REV_NUM

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_DOC_QUOTE_BU	DOC_QUOTE_NAME	[DOC_QUOTE_ID].[NAME]	S_DOC_QUOTE	NAME
S_INSCLM_BU	INSCLAIM_NUM	[INSCLM_ID].[INSCLAIM_NUM]	S_INSCLAIM	INSCLAIM_NUM
S_INSCLM_BU	REV_NUM	[INSCLM_ID].[REVISION_NUM]	S_INSCLAIM	REVISION_NUM
S_INSCLMEL_BU	SEQ_NUM	[INSCLM_ELMNT_ID].[SEQ_NUM]	S_INSCLM_ELMNT	SEQ_NUM
S_INSCLMEL_BU	TYPE_CD	[INSCLM_ELMNT_ID].[TYPE_CD]	S_INSCLM_ELMNT	TYPE_CD
S_INV_TXN_BU	INV_TXN_NUM	[INV_TXN_ID].[INV_TXN_NUM]	S_INV_TXN	INV_TXN_NUM
S_ISS_BU	ISS_LANG_ID	[ISS_ID].[LANG_ID]	S_ISS	LANG_ID
S_ISS_BU	ISS_NAME	[ISS_ID].[NAME]	S_ISS	NAME
S_LOY_MEM_BU	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_BU	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_MEM_PSTN	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_PSTN	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_PROG_BU	PROG_NAME	[PROG_ID].[NAME]	S_LOY_PROG	NAME
S_LOY_PROMO_BU	PROMO_NAME	[PROMO_ID].[NAME]	S_LOY_PROMO	NAME
S_LOY_PROMO_BU	PROMO_NUM	[PROMO_ID].[PROMO_NUM]	S_LOY_PROMO	PROMO_NUM
S_LOY_TXN_BU	TXN_NUM	[TXN_ID].[TXN_NUM]	S_LOY_TXN	TXN_NUM
S_LOY_TXN_BU	TXN_STATUS_CD	[TXN_ID].[STATUS_CD]	S_LOY_TXN	STATUS_CD
S_LOY_TXN_BU	TXN_SUB_TYPE_CD	[TXN_ID].[SUB_TYPE_CD]	S_LOY_TXN	SUB_TYPE_CD
S_LOY_TXN_BU	TXN_TYPE_CD	[TXN_ID].[TYPE_CD]	S_LOY_TXN	TYPE_CD

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_MDF_BU	MDF_UI D	[MDF_ID].[MDF_UI D]	S_MDF	MDF_UI D
S_OPTY_BU	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME
S_OPTY_BU	SUM_CLASS_CD	[OPTY_ID].[SUM_C LASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_BU	SUM_COMMIT_FLG	[OPTY_ID].[SUM_C OMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG
S_OPTY_BU	SUM_COST_AMT	[OPTY_ID].[SUM_C OST_AMT]	S_OPTY	SUM_COST_AMT
S_OPTY_BU	SUM_DOWNSIDE_A MT	[OPTY_ID].[SUM_D OWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_A MT
S_OPTY_BU	SUM_EFFECTIVE_ DT	[OPTY_ID].[SUM_E FFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_ DT
S_OPTY_BU	SUM_MARGIN_AMT	[OPTY_ID].[SUM_M ARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_BU	SUM_REVN_AMT	[OPTY_ID].[SUM_R EVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_BU	SUM_TYPE_CD	[OPTY_ID].[SUM_T YPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_BU	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_U PSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_BU	SUM_WIN_PROB	[OPTY_ID].[SUM_W IN_PROB]	S_OPTY	SUM_WIN_PROB
S_OPTY_POSTN	CONSUMER_OPTY_ FLG	[OPTY_ID].[CONSU MER_OPTY_FLG]	S_OPTY	CONSUMER_OPTY_ FLG
S_OPTY_POSTN	NEW_LOAN_FLG	[OPTY_ID].[NEW_L OAN_FLG]	S_OPTY	NEW_LOAN_FLG
S_OPTY_POSTN	OPTY_CLOSED_FL G	[OPTY_ID].[CLOSE D_FLG]	S_OPTY	CLOSED_FLG
S_OPTY_POSTN	OPTY_NAME	[OPTY_ID].[NAME]	S_OPTY	NAME
S_OPTY_POSTN	SECURE_FLG	[OPTY_ID].[SECU RE_FLG]	S_OPTY	SECURE_FLG
S_OPTY_POSTN	SUM_CLASS_CD	[OPTY_ID].[SUM_C LASS_CD]	S_OPTY	SUM_CLASS_CD
S_OPTY_POSTN	SUM_COMMIT_FLG	[OPTY_ID].[SUM_C OMMIT_FLG]	S_OPTY	SUM_COMMIT_FLG

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_OPTY_POSTN	SUM_COST_AMT	[OPTY_ID].[SUM_COST_AMT]	S_OPTY	SUM_COST_AMT
S_OPTY_POSTN	SUM_DOWNSIDE_AMT	[OPTY_ID].[SUM_DOWNSIDE_AMT]	S_OPTY	SUM_DOWNSIDE_AMT
S_OPTY_POSTN	SUM_EFFECTIVE_DT	[OPTY_ID].[SUM_EFFECTIVE_DT]	S_OPTY	SUM_EFFECTIVE_DT
S_OPTY_POSTN	SUM_MARGIN_AMT	[OPTY_ID].[SUM_MARGIN_AMT]	S_OPTY	SUM_MARGIN_AMT
S_OPTY_POSTN	SUM_REVN_AMT	[OPTY_ID].[SUM_REVN_AMT]	S_OPTY	SUM_REVN_AMT
S_OPTY_POSTN	SUM_TYPE_CD	[OPTY_ID].[SUM_TYPE_CD]	S_OPTY	SUM_TYPE_CD
S_OPTY_POSTN	SUM_UPSIDE_AMT	[OPTY_ID].[SUM_UPSIDE_AMT]	S_OPTY	SUM_UPSIDE_AMT
S_OPTY_POSTN	SUM_WIN_PROB	[OPTY_ID].[SUM_WIN_PROB]	S_OPTY	SUM_WIN_PROB
S_ORDER_BU	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_BU	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_BU	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORDER_POSTN	ORDER_CAT_CD	[ORDER_ID].[ORDER_CAT_CD]	S_ORDER	ORDER_CAT_CD
S_ORDER_POSTN	ORDER_DT	[ORDER_ID].[ORDER_DT]	S_ORDER	ORDER_DT
S_ORDER_POSTN	ORDER_NUM	[ORDER_ID].[ORDER_NUM]	S_ORDER	ORDER_NUM
S_ORG_BU	ORG_CMPT_FLG	[ORG_ID].[CMPT_FLG]	S_ORG_EXT	CMPT_FLG
S_ORG_BU	ORG_FACILITY_FLG	[ORG_ID].[FACILITY_FLG]	S_ORG_EXT	FACILITY_FLG
S_ORG_BU	ORG_INVSTR_FLG	[ORG_ID].[INVSTR_FLG]	S_ORG_EXT	INVSTR_FLG
S_ORG_BU	ORG_LOC	[ORG_ID].[LOC]	S_ORG_EXT	LOC
S_ORG_BU	ORG_NAME	[ORG_ID].[NAME]	S_ORG_EXT	NAME

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ORG_BU	ORG_PRTNR_FLG	[ORG_ID].[PRTNR_FLG]	S_ORG_EXT	PRTNR_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_BU	ORG_SRV_PROVDR_FLG	[ORG_ID].[SRV_PROVIDR_FLG]	S_ORG_EXT	SRV_PROVIDR_FLG
S_ORG_DIST_LST	OU_ID	[ORG_PROD_ID].[OU_ID]	S_ORG_PROD	OU_ID
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_ORG_GROUP_BU	OG_GROUP_TYPE_CD	[ORG_GROUP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GROUP_TYPE_CD	[ORGGRP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_ORGGRP_POSTN	GRP_NAME	[ORGGRP_ID].[NAME]	S_ORG_GROUP	NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_PERIOD_BU	PERIOD_NAME	[PERIOD_ID].[NAME]	S_PERIOD	NAME
S_POS_BU	POS_NUM	[POS_ID].[POS_NUM]	S_POS	POS_NUM
S_POSTN_CON	AGENT_FLG	[CON_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_POSTN_CON	MEMBER_FLG	[CON_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_POSTN_CON	PROVIDER_FLG	[CON_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_PRIMFCTR_ITM	PRIMDL_ID	[PRIMDL_FCTR_ID].[PRIMDL_ID]	S_PRIMDL_FCTR	PRIMDL_ID
S_PROD_DFCT_BU	DEFECT_NUM	[PROD_DEFECT_ID].[DEFECT_NUM]	S_PROD_DEFECT	DEFECT_NUM

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_PROD_INT_BU	PROD_CD	[PROD_INT_ID].[PROD_CD]	S_PROD_INT	PROD_CD
S_PROD_INT_BU	PROD_EFF_END_DT	[PROD_INT_ID].[EFF_END_DT]	S_PROD_INT	EFF_END_DT
S_PROD_INT_BU	PROD_EFF_START_DT	[PROD_INT_ID].[EFF_START_DT]	S_PROD_INT	EFF_START_DT
S_PROD_STYL_TNT	SETUP_STYLE_CD	[PROP_STYLE_ID].[SETUP_STYLE_CD]	S_PROP_STYL_TNT	SETUP_STYLE_CD
S_PROJ_BU	PROJ_NAME	[PROJ_ID].[NAME]	S_PROJ	NAME
S_PROJ_BU	PROJ_STATUS_CD	[PROJ_ID].[STATUS_CD]	S_PROJ	STATUS_CD
S_PROJ_BU	PROJ_TYPE_CD	[PROJ_ID].[PROJ_TYPE_CD]	S_PROJ	PROJ_TYPE_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SALES_MTHD_BU	SALES_METHOD_NAME	[SALES_METHOD_ID].[NAME]	S_SALES_METHOD	NAME
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD

Table 51. Columns Denormalized During Upgrades from Release 6.3 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_TMPL_PLNIT_BU	PLANITEM_NAME	[TMPL_PLANITEM_ID].[NAME]	S_TMPL_PLANITEM	NAME
S_TMPL_PLNIT_BU	PLANITEM_TYPE_CD	[TMPL_PLANITEM_ID].[TYPE_CD]	S_TMPL_PLANITEM	TYPE_CD
S_TMPL_PLNIT_BU	PLNIT_TMPL_TYPE_CD	[TMPL_PLANITEM_ID].[TMPL_TYPE_CD]	S_TMPL_PLANITEM	TMPL_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
S_WRNTY_CVRG_BU	WRNTY_CVRG_NAME	[WRNTY_CVRG_ID].[WRNTY_NAME]	S_WRNTY_CVRG	WRNTY_NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 7.0.4 Siebel eBusiness Applications

Table 52 lists columns that are denormalized during upgrades from Release 7.0.4 Siebel eBusiness applications to Release 7.8 Siebel Industry applications.

Table 52. Columns Denormalized During Upgrades from Release 7.0.4 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ORG_REF_CUST_FLG	[OU_EXT_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_CAL_RSRC	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_CAL_RSRC	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_CAL_RSRC	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT
S_ACT_CAL_RSRC	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_ACT_EMP	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_CONTACT_BU	CON_FST_NAME	[CONTACT_ID].[FST_NAME]	S_CONTACT	FST_NAME
S_CONTACT_BU	CON_LAST_NAME	[CONTACT_ID].[LAST_NAME]	S_CONTACT	LAST_NAME
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME

Table 52. Columns Denormalized During Upgrades from Release 7.0.4 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME

Table 52. Columns Denormalized During Upgrades from Release 7.0.4 Siebel eBusiness Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 7.0.4 Siebel Financial Services Applications

Table 53 lists columns that are denormalized during upgrades from Release 7.0.4 Siebel Financial Services applications to Release 7.8 Siebel Industry applications.

Table 53. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ACCNT_TYPE_CD	[OU_EXT_ID].[ACCNT_TYPE_CD]	S_ORG_EXT	ACCNT_TYPE_CD
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_CAL_RSRC	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_CAL_RSRC	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_CAL_RSRC	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT
S_ACT_CAL_RSRC	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_ACT_EMP	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ASSET_BU	ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM

Table 53. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ASSET_BU	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD
S_CASE_BU	CASE_NAME	[CASE_ID].[NAME]	S_CASE	NAME
S_CASE_BU	CASE_STATUS_CD	[CASE_ID].[STATUS_CD]	S_CASE	STATUS_CD
S_CASE_BU	CASE_TYPE_CD	[CASE_ID].[TYPE_CD]	S_CASE	TYPE_CD
S_CONTACT_BU	AGENT_FLG	[CONTACT_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_CONTACT_BU	CON_FST_NAME	[CONTACT_ID].[FST_NAME]	S_CONTACT	FST_NAME
S_CONTACT_BU	CON_LAST_NAME	[CONTACT_ID].[LAST_NAME]	S_CONTACT	LAST_NAME
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	MEMBER_FLG	[CONTACT_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_CONTACT_BU	PROVIDER_FLG	[CONTACT_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_INSCLM_BU	INSCCLAIM_NUM	[INSCLM_ID].[INSCCLAIM_NUM]	S_INSCCLAIM	INSCCLAIM_NUM
S_INSCLM_BU	REV_NUM	[INSCLM_ID].[REVISION_NUM]	S_INSCCLAIM	REVISION_NUM
S_INSCLMEL_BU	SEQ_NUM	[INSCLM_ELMNT_ID].[SEQ_NUM]	S_INSCLM_ELMNT	SEQ_NUM
S_INSCLMEL_BU	TYPE_CD	[INSCLM_ELMNT_ID].[TYPE_CD]	S_INSCLM_ELMNT	TYPE_CD
S_LOY_MEM_BU	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_BU	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_MEM_PSTN	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM

Table 53. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_LOY_MEM_PSTN	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_PROG_BU	PROG_NAME	[PROG_ID].[NAME]	S_LOY_PROGRAM	NAME
S_LOY_PROMO_BU	PROMO_NAME	[PROMO_ID].[NAME]	S_LOY_PROMO	NAME
S_LOY_PROMO_BU	PROMO_NUM	[PROMO_ID].[PROMO_NUM]	S_LOY_PROMO	PROMO_NUM
S_LOY_TXN_BU	TXN_NUM	[TXN_ID].[TXN_NUM]	S_LOY_TXN	TXN_NUM
S_LOY_TXN_BU	TXN_STATUS_CD	[TXN_ID].[STATUS_CD]	S_LOY_TXN	STATUS_CD
S_LOY_TXN_BU	TXN_SUB_TYPE_CD	[TXN_ID].[SUB_TYPE_CD]	S_LOY_TXN	SUB_TYPE_CD
S_LOY_TXN_BU	TXN_TYPE_CD	[TXN_ID].[TYPE_CD]	S_LOY_TXN	TYPE_CD
S_OPTY_POSTN	NEW_LOAN_FLG	[OPTY_ID].[NEW_LOAN_FLG]	S_OPTY	NEW_LOAN_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_DIST_LST	OU_ID	[ORG_PROD_ID].[OU_ID]	S_ORG_PROD	OU_ID
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_ORG_GROUP_BU	OG_GROUP_TYPE_CD	[ORG_GROUP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_PROD_STYL_TNT	SETUP_STYLE_CD	[PROP_STYLE_ID].[SETUP_STYLE_CD]	S_PROP_STYL_TNT	SETUP_STYLE_CD
S_PROJ_BU	PROJ_NAME	[PROJ_ID].[NAME]	S_PROJ	NAME

Table 53. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_PROJ_BU	PROJ_STATUS_CD	[PROJ_ID].[STATUS_CD]	S_PROJ	STATUS_CD
S_PROJ_BU	PROJ_TYPE_CD	[PROJ_ID].[PROJ_TYPE_CD]	S_PROJ	PROJ_TYPE_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Table 53. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Financial Services Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 7.0.4 Siebel Industry Solutions

Table 54 lists columns that are denormalized during upgrades from Release 7.0.4 Siebel Industry Solutions to Release 7.8 Siebel Industry applications.

Table 54. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACCNT_POSTN	ACCNT_TYPE_CD	[OU_EXT_ID].[ACCNT_TYPE_CD]	S_ORG_EXT	ACCNT_TYPE_CD
S_ACCNT_POSTN	CLIENT_FLG	[OU_EXT_ID].[CLIENT_FLG]	S_ORG_EXT	CLIENT_FLG
S_ACCNT_POSTN	FACILITY_FLG	[OU_EXT_ID].[FACILITY_FLG]	S_ORG_EXT	FACILITY_FLG
S_ACCNT_POSTN	INVSTR_FLG	[OU_EXT_ID].[INVSTR_FLG]	S_ORG_EXT	INVSTR_FLG
S_ACCNT_POSTN	ORG_REF_CUST_FLG	[OU_EXT_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ACCNT_POSTN	SRV_PROVDR_FLG	[OU_EXT_ID].[SRV_PROVDR_FLG]	S_ORG_EXT	SRV_PROVDR_FLG
S_ACT_CAL_RSRC	ACT_APPT_REPT_FLG	[ACTIVITY_ID].[APPT_REPT_FLG]	S_EVT_ACT	APPT_REPT_FLG
S_ACT_CAL_RSRC	ACT_APPT_RPTEND_DT	[ACTIVITY_ID].[APPT_REPT_END_DT]	S_EVT_ACT	APPT_REPT_END_DT
S_ACT_CAL_RSRC	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ACT_CAL_RSRC	ACT_TEMPLATE_FLG	[ACTIVITY_ID].[TEMPLATE_FLG]	S_EVT_ACT	TEMPLATE_FLG
S_ACT_CAL_RSRC	ACT_TODO_PLNEND_DT	[ACTIVITY_ID].[TODO_PLAN_END_DT]	S_EVT_ACT	TODO_PLAN_END_DT

Table 54. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ACT_CAL_RSRC	ACT_TODO_PLNSTRTDT	[ACTIVITY_ID].[TODO_PLAN_START_DT]	S_EVT_ACT	TODO_PLAN_START_DT
S_ACT_EMP	ACT_CAL_TYPE_CD	[ACTIVITY_ID].[CAL_TYPE_CD]	S_EVT_ACT	CAL_TYPE_CD
S_ASSET_BU	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD
S_ASSET_POSTN	ASSET_NUM	[ASSET_ID].[ASSET_NUM]	S_ASSET	ASSET_NUM
S_ASSET_POSTN	TYPE_CD	[ASSET_ID].[TYPE_CD]	S_ASSET	TYPE_CD
S_CASE_BU	CASE_NAME	[CASE_ID].[NAME]	S_CASE	NAME
S_CASE_BU	CASE_STATUS_CD	[CASE_ID].[STATUS_CD]	S_CASE	STATUS_CD
S_CASE_BU	CASE_TYPE_CD	[CASE_ID].[TYPE_CD]	S_CASE	TYPE_CD
S_CONTACT_BU	AGENT_FLG	[CONTACT_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_CONTACT_BU	CON_EMP_FLG	[CONTACT_ID].[EMP_FLG]	S_CONTACT	EMP_FLG
S_CONTACT_BU	CON_MID_NAME	[CONTACT_ID].[MID_NAME]	S_CONTACT	MID_NAME
S_CONTACT_BU	MEMBER_FLG	[CONTACT_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_CONTACT_BU	PROVIDER_FLG	[CONTACT_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_INSCLM_BU	INSCCLAIM_NUM	[INSCLM_ID].[INSCCLAIM_NUM]	S_INSCCLAIM	INSCCLAIM_NUM
S_INSCLM_BU	REV_NUM	[INSCLM_ID].[REVISION_NUM]	S_INSCCLAIM	REVISION_NUM
S_INSCLMEL_BU	SEQ_NUM	[INSCLM_ELMNT_ID].[SEQ_NUM]	S_INSCLM_ELMNT	SEQ_NUM
S_INSCLMEL_BU	TYPE_CD	[INSCLM_ELMNT_ID].[TYPE_CD]	S_INSCLM_ELMNT	TYPE_CD
S_LOY_MEM_BU	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM

Table 54. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_LOY_MEM_BU	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_MEM_PSTN	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_PSTN	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_PROG_BU	PROG_NAME	[PROG_ID].[NAME]	S_LOY_PROGRAM	NAME
S_LOY_PROMO_BU	PROMO_NAME	[PROMO_ID].[NAME]	S_LOY_PROMO	NAME
S_LOY_PROMO_BU	PROMO_NUM	[PROMO_ID].[PROMO_NUM]	S_LOY_PROMO	PROMO_NUM
S_LOY_TXN_BU	TXN_NUM	[TXN_ID].[TXN_NUM]	S_LOY_TXN	TXN_NUM
S_LOY_TXN_BU	TXN_STATUS_CD	[TXN_ID].[STATUS_CD]	S_LOY_TXN	STATUS_CD
S_LOY_TXN_BU	TXN_SUB_TYPE_CD	[TXN_ID].[SUB_TYPE_CD]	S_LOY_TXN	SUB_TYPE_CD
S_LOY_TXN_BU	TXN_TYPE_CD	[TXN_ID].[TYPE_CD]	S_LOY_TXN	TYPE_CD
S_OPTY_POSTN	CONSUMER_OPTY_FLG	[OPTY_ID].[CONSUMER_OPTY_FLG]	S_OPTY	CONSUMER_OPTY_FLG
S_OPTY_POSTN	NEW_LOAN_FLG	[OPTY_ID].[NEW_LOAN_FLG]	S_OPTY	NEW_LOAN_FLG
S_OPTY_POSTN	OPTY_CLOSED_FLG	[OPTY_ID].[CLOSED_FLG]	S_OPTY	CLOSED_FLG
S_OPTY_POSTN	SECURE_FLG	[OPTY_ID].[SECURE_FLG]	S_OPTY	SECURE_FLG
S_ORG_BU	ORG_FACILITY_FLG	[ORG_ID].[FACILITY_FLG]	S_ORG_EXT	FACILITY_FLG
S_ORG_BU	ORG_INVSTR_FLG	[ORG_ID].[INVSTR_FLG]	S_ORG_EXT	INVSTR_FLG
S_ORG_BU	ORG_REF_CUST_FLG	[ORG_ID].[REFERENCE_CUST_FLG]	S_ORG_EXT	REFERENCE_CUST_FLG
S_ORG_BU	ORG_SRV_PROVDR_FLG	[ORG_ID].[SRV_PROVDR_FLG]	S_ORG_EXT	SRV_PROVDR_FLG

Table 54. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_ORG_DIST_LST	OU_ID	[ORG_PROD_ID].[OU_ID]	S_ORG_PROD	OU_ID
S_ORG_GROUP_BU	OG_GROUP_NAME	[ORG_GROUP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_ORG_GROUP_BU	OG_GROUP_TYPE_CD	[ORG_GROUP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORG_GROUP_BU	OG_NAME	[ORG_GROUP_ID].[NAME]	S_ORG_GROUP	NAME
S_ORGGRP_POSTN	GROUP_TYPE_CD	[ORGGRP_ID].[GROUP_TYPE_CD]	S_ORG_GROUP	GROUP_TYPE_CD
S_ORGGRP_POSTN	GRP_GROUP_NAME	[ORGGRP_ID].[GROUP_NAME]	S_ORG_GROUP	GROUP_NAME
S_PARTY_GRP_BU	PG_NAME	[PARTY_GROUP_ID].[NAME]	S_PARTY_GROUP	NAME
S_POS_BU	POS_NUM	[POS_ID].[POS_NUM]	S_POS	POS_NUM
S_POSTN_CON	AGENT_FLG	[CON_ID].[AGENT_FLG]	S_CONTACT	AGENT_FLG
S_POSTN_CON	MEMBER_FLG	[CON_ID].[MEMBER_FLG]	S_CONTACT	MEMBER_FLG
S_POSTN_CON	PROVIDER_FLG	[CON_ID].[PROVIDER_FLG]	S_CONTACT	PROVIDER_FLG
S_PROD_INT_BU	PROD_CD	[PROD_INT_ID].[PROD_CD]	S_PROD_INT	PROD_CD
S_PROD_STYL_TNT	SETUP_STYLE_CD	[PROP_STYLE_ID].[SETUP_STYLE_CD]	S_PROP_STYL_TNT	SETUP_STYLE_CD
S_PROJ_BU	PROJ_NAME	[PROJ_ID].[NAME]	S_PROJ	NAME
S_PROJ_BU	PROJ_STATUS_CD	[PROJ_ID].[STATUS_CD]	S_PROJ	STATUS_CD
S_PROJ_BU	PROJ_TYPE_CD	[PROJ_ID].[PROJ_TYPE_CD]	S_PROJ	PROJ_TYPE_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM
S_SRC_BU	SRC_NAME	[SRC_ID].[NAME]	S_SRC	NAME

Table 54. Columns Denormalized During Upgrades from Release 7.0.4 Siebel Industry Solutions

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_SRV_REQ_BU	SR_AREA	[SRV_REQ_ID].[SR_AREA]	S_SRV_REQ	SR_AREA
S_SRV_REQ_BU	SR_CST_NUM	[SRV_REQ_ID].[SR_CST_NUM]	S_SRV_REQ	SR_CST_NUM
S_SRV_REQ_BU	SR_NUM	[SRV_REQ_ID].[SR_NUM]	S_SRV_REQ	SR_NUM
S_SRV_REQ_BU	SR_SEV_CD	[SRV_REQ_ID].[SR_SEV_CD]	S_SRV_REQ	SR_SEV_CD
S_SRV_REQ_BU	SR_STAT_ID	[SRV_REQ_ID].[SR_STAT_ID]	S_SRV_REQ	SR_STAT_ID
S_SRV_REQ_BU	SR_SUB_STAT_ID	[SRV_REQ_ID].[SR_SUB_STAT_ID]	S_SRV_REQ	SR_SUB_STAT_ID
S_SRV_REQ_BU	SR_TITLE	[SRV_REQ_ID].[SR_TITLE]	S_SRV_REQ	SR_TITLE
S_SRV_REQ_BU	SR_TYPE_CD	[SRV_REQ_ID].[SR_TYPE_CD]	S_SRV_REQ	SR_TYPE_CD
S_USERLIST_BU	UL_NAME	[USERLIST_ID].[NAME]	S_USERLIST	NAME
T_MASTER_BU	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_BU	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_DETAIL	DETAIL_SX2_NAME	[DETAIL_SX2_ID].[SX2_NAME]	T_DETAIL_SX2	SX2_NAME
T_MASTER_DETAIL	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME
T_MASTER_PER	MASTER_ALIAS	[MASTER_ID].[ALIAS]	T_MASTER	ALIAS
T_MASTER_POSTN	MASTER_NAME	[MASTER_ID].[NAME]	T_MASTER	NAME

Denormalized Columns for 7.5.2 Siebel Industry Applications

Table 55 lists columns that are denormalized during upgrades from Release 7.5.2 Siebel Industry applications to Release 7.8 Siebel Industry applications.

Table 55. Columns Denormalized During Upgrades from Release 7.5.2 Siebel Industry Applications

Target Table	Target Column	Denorm Path	Source Table	Source Column
S_LOY_MEM_BU	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_BU	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_MEM_PSTN	MEM_NUM	[MEMBER_ID].[MEM_NUM]	S_LOY_MEMBER	MEM_NUM
S_LOY_MEM_PSTN	MEM_TYPE_CD	[MEMBER_ID].[MEM_TYPE_CD]	S_LOY_MEMBER	MEM_TYPE_CD
S_LOY_PROG_BU	PROG_NAME	[PROG_ID].[NAME]	S_LOY_PROGRAM	NAME
S_LOY_PROMO_BU	PROMO_NAME	[PROMO_ID].[NAME]	S_LOY_PROMO	NAME
S_LOY_PROMO_BU	PROMO_NUM	[PROMO_ID].[PROMO_NUM]	S_LOY_PRO	PROMO_NUM
S_LOY_TXN_BU	TXN_NUM	[TXN_ID].[TXN_NUM]	S_LOY_TXN	TXN_NUM
S_LOY_TXN_BU	TXN_STATUS_CD	[TXN_ID].[STATUS_CD]	S_LOY_TXN	STATUS_CD
S_LOY_TXN_BU	TXN_SUB_TYPE_CD	[TXN_ID].[SUB_TYPE_CD]	S_LOY_TXN	SUB_TYPE_CD
S_LOY_TXN_BU	TXN_TYPE_CD	[TXN_ID].[TYPE_CD]	S_LOY_TXN	TYPE_CD
S_POS_BU	POS_NUM	[POS_ID].[POS_NUM]	S_POS	POS_NUM
S_PROD_STYL_TNT	SETUP_STYLE_CD	[PROP_STYLE_ID].[SETUP_STYLE_CD]	S_PROP_STYL_TNT	SETUP_STYLE_CD
S_PSP_PROC_BU	VOD_NAME	[VOD_ID].[VOD_NAME]	S_VOD	VOD_NAME
S_QUOTE_POSTN	QUOTE_NUM	[QUOTE_ID].[QUOTE_NUM]	S_DOC_QUOTE	QUOTE_NUM

E

Production Upgrade Files Generated by the Upgrade Wizard

This appendix lists the files that are used to perform the upgrade to Release 7.8. This appendix also lists the tables amended during PRET (pre-table) table processing.

This appendix contains the following topics:

- [“Siebel 7.8 Upgrade Files”](#)
- [“Tables Amended During PRET Unload Processing” on page 465](#)
- [“PRET Members Generated By Pretedit.txt” on page 467](#)
- [“Source Tables Amended During PRET Processing” on page 468](#)

Siebel 7.8 Upgrade Files

When you run the Siebel Upgrade Wizard on the midtier, it generates the files that are used to perform the production upgrade. You then transfer these files to the z/OS host, where they are placed in staging data sets. Several upgrade files are also generated on the z/OS host. [Table 56](#) lists each of the production upgrade files generated on the midtier, the staging data set name assigned to the file when it is transferred to the z/OS host, the phase of the upgrade when the file is generated, the file type, and a brief description. The upgrade path the file applies to is also listed.

Table 56. Production Upgrade Files

Midtier File Name	Staging Data Set Name	Phase	File Type	Description	Applicable Upgrade Paths
job0.txt	JOB0	Prep	iebudpte	REXX code and panels	All
job1.txt	VSTG0002	Prep	iebudpte	Generic install.jcl and help panels	All
	VSTG0020				
	VSTG0021				
	VSTG0022				
	VSTG0003				
	VSTG0005				
	VSTG0006				
	VSTG0040				
	VSTG0041				

Table 56. Production Upgrade Files

Midtier File Name	Staging Data Set Name	Phase	File Type	Description	Applicable Upgrade Paths
siebin01.jcl	VSTG0011	Prep	iebupdte	Install.jcl specific to each upgrade path	All
	VSTG0030				
	VSTG0031				
	VSTG0032				
	VSTG0012				
	VSTG0014				
	VSTG0015				
	VSTG0050				
VSTG0051					
siebproc.jcl	VSTG0070	Prep	iebupdte	JCL PROC members	All
filelist.txt	VSTG0075			List of files	All
dedup.jcl	VSTG0080	Maintenance	SQL	ITM/Dedup Remote conflict_id process	All
dedup_prod.jcl	VSTG0081	Maintenance	JCL	ITM/Dedup Remote conflict_id process	All
schema.db.sql	VSTG0100		DDL	Schema databases	All
schema.tbsp.sql	VSTG0101		DDL	Schema table spaces	All
schema.tbl.sql	VSTG0102		DDL	Schema tables	All
schema.grt.sql	VSTG0103		DDL	Schema grants	All
schema.uind.sql	VSTG0104		DDL	Non-obsolete and unique index	All
schema.nuind.sql	VSTG0105		DDL	Non-obsolete and non-unique index	All
schema.oind.sql	VSTG0106		DDL	Obsolete indexes	All
scindx.sql	VSTG0110		DDL	File is split into: Drops, EIM, Pre/Post Gen_Primary DDL and Indexes rebuilds	All

Table 56. Production Upgrade Files

Midtier File Name	Staging Data Set Name	Phase	File Type	Description	Applicable Upgrade Paths
tmptable.sql	VSTG0120		DDL	Common TMPTABLES (tmptable.ctf)	All
tmptable1.sql	VSTG0121		DDL	Source table logging for Unload jobs (no storage.ctf file)	All
tmptable2.sql	VSTG0122		DDL	Target table logging for Unload jobs (no storage.ctf file)	All
ddlviews.sql	VSTG0130		DDL		All
infdrops.sql	VSTG0140		SQL	Interface/EIM Drops - part of pretftp	
siebel.translate.iconv	VSTG0150		binary	Program to convert code pages for each language	All
pretdit.txt	VSTG0200	DM-Prep	JCL (pretldin, pretclob)	PRET JCL	All
pret.jcl	VSTG0210	DM-Prep	SQL	PRET SQL	All
pret_prod.jcl	VSTG0211	DM-Prep	JCL	PRET JCL	All
pret_sia.jcl	VSTG0220	DM-Prep	SQL	PRET SQL	SIA621, SIA704, SIA752, SIA77
pret_prod_sia.jcl	VSTG0221	DM-Prep	JCL	PRET JCL	SIA621, SIA704, SIA752, SIA77
unload.ldc	VSTG0300	DDLIMP	LDC	Unload control cards	All
load.ldc	VSTG0310	DDLIMP	LDC	Load control cards	All
preschm.jcl	VSTG0400	DM-Horz	SQL	PRESCHM SQL	All
preschm_prod.jcl	VSTG0401	DM-Horz	JCL	PRESCHM JCL	All
preschm.ldc	VSTG0402	DM-Horz	LDC	Load control cards for PRESCHM	SIA621
preschm_sia.jcl	VSTG0410	DM-SIA	SQL	PRESCHM SQL	SIA621, SIA704, SIA752, SIA77

Table 56. Production Upgrade Files

Midtier File Name	Staging Data Set Name	Phase	File Type	Description	Applicable Upgrade Paths
preschm_sia_prod.jcl	VSTG0411	DM-SIA	JCL	PRESCHM JCL	SIA621, SIA704, SIA752, SIA77
preschm_sia.ldc	VSTG0412	DM-SIA	LDC		
prod_confirgurator.jcl	VSTG0500	DM-Horz	SQL	Prod Configurator SQL	SIA621
prod_confirgurator_prod.jcl	VSTG0501	DM-Horz	JCL	Prod Configurator JCL	SIA621
prod_confirgurator.ldc	VSTG0502	DM-Horz	LDC		SIA621
prod_confirgurator_sia.jcl	VSTG0510	DM-SIA	SQL	Prod Configurator SQL	SIA621, SIA704
prod_confirgurator_sia_prod.jcl	VSTG0511	DM-SIA	JCL	Prod Configurator JCL	SIA621, SIA704
prod_confirgurator_sia.ldc	VSTG0512	DM-SIA	LDC		SIA621, SIA704
upg_iss.jcl	VSTG0600	DM	JCL	UPGISS JCL	All
upg_iss_prod.jcl	VSTG0601	DM	SQL	UPGISS SQL	All
upg_iss.ldc	VSTG0602	DM	LDC		All
gen_primary1.jcl	VSTG0700	DM	SQL	Gen primary part 1 - SQL	All
gen_primary1_prod.jcl	VSTG0701	DM	JCL	Gen primary part 1 - JCL	All
gen_primary2.jcl	VSTG0702	DM	SQL	Gen primary part 2 - SQL	All
gen_primary2_prod.jcl	VSTG0703	DM	JCL	Gen primary part 2 - JCL	All
gen_primary3.jcl	VSTG0704	DM	SQL	Gen primary part 3 - SQL	All
gen_primary3_prod.jcl	VSTG0705	DM	JCL	Gen primary part 3 - JCL	All
gen_primary4.jcl	VSTG0706	DM	SQL	Gen primary part 4 - SQL	All

Table 56. Production Upgrade Files

Midtier File Name	Staging Data Set Name	Phase	File Type	Description	Applicable Upgrade Paths
gen_primary4_prod.jcl	VSTG0707	DM	JCL	Gen primary part 4 - JCL	All
echannel_merge_contact.sql	VSTG0800	DM	SQL	eChannel	SIA621
echannel_merge_org_int.sql	VSTG0801	DM	SQL	eChannel	SIA621
echannelAccountDivMatch_fins.sql	Mid-tier	DM	SQL	eChannel	SIA621
echannelEmpContactMatch_fins.sql	Mid-tier	DM	SQL	eChannel	SIA621
hhmignot.sql	Mid-tier	DM	SQL	Household	SIA621, SIA704, SIA752, SIA77
hhmigpop.sql	Mid-tier	DM	SQL	Household	SIA621, SIA704, SIA752, SIA77
Household_Mig_Fins.jcl	VSTG0852	DM	SQL	Household	SIA621, SIA704, SIA752, SIA77
Household_Mig_Fins_prod.jcl	VSTG0853	DM	JCL	Household	SIA621, SIA704, SIA752, SIA77
Household_Mig_Fins_prod.ldc	VSTG0854	DM	LDC	Household	SIA621, SIA704, SIA752, SIA77
rpt_dup_addr_ro_wids.sql	Mid-tier	DM	SQL	Gen Dup Addr Report SQL	SIA621, SIA704, SIA752, SIA77
rpt_dup_addr_names.sql	VSTG0860	DM	SQL	Gen Dup Addr Report SQL	All

Tables Amended During PRET Unload Processing

The unload job control cards for specific source tables have been modified so that during PRET (pre-table) processing, the data in the tables is modified during the table unload process instead of being modified after the data has been loaded into the target table.

Table 57 lists the source tables containing the data that is modified during unload processing, the macro that performs the modifications, and the relevant upgrade paths.

Table 57. Tables Containing Data Modified During the PRET Table Unload Process

PRET Tables Modified During the Table Unload Process	Macro	Hor Paths	SIA Paths
S_APPL_WEB_TMPL	PTH0062	HOR704	SIA704, SIA752
S_APPL_WTMPL_IT	PTH0064	HOR704	SIA704, SIA752
S_POSTN_CON	PTS0100	n/a	SIA704 (FINS704 only)
S_CONTROL	PTH0222	All	All
S_PCONTROL	PTS0223	HOR752, HOR77	SIA752
S_FN_CRDT_RPT	PTS0224	HOR752, HOR77	SIA752
S_ASGN_RULE_GRP	PTS0225	HOR752, HOR77	SIA752
S_REGION	PTS0227	n/a	SIA 704 (FINS704 only), SIA752
S_REGION	PTS0228	n/a	SIA752
S_ENTLMNT_ITEM_FEE	PTS0270 PTS0271	n/a	SIS704
S_CURRCLM_PER	PTH0289	HOR621	SIA621, SIS63
S_EVT_ACT	PTH0405	HOR621	SIA621, SIS63
S_QTA_OBJCRT	PTH0832	HOR621	SIA621, SIS63
S_QTA_OBJCRT_D	PTH0833	All	All
S_WF_PROC_FLOW	PTH1003	HOR621, HOR704	SIA621, SIS63, SIA704
S_WF_STEP	PTH1006	HOR621, HOR704	SIA621, SIS63, SIA704
S_ASSET_POSTN	PTS0301 PTS0302 PTS0303 PTS0304	n/a	SIS 704 only
S_ETL_TIME_DAY	PTS0313 PTS0314	n/a	SIA752
S_DD_MEAS_ATTR	PTS0350	n/a	SIS63
S_UPG_KIT_IARG	PTS0360 PTS0361	HOR704, HOR752, HOR77	SIA704(SIS + FINS), SIA752, SIA77
S_CON_ADDR	PTM0410	n/a	SIA621, SIS63, SIA704
S_EXTDATA_TBL	PTS0500	n/a	SIA752
S_PAPL_WEB_TMPL	PTS0505	n/a	SIA752

Table 57. Tables Containing Data Modified During the PRET Table Unload Process

PRET Tables Modified During the Table Unload Process	Macro	Hor Paths	SIA Paths
S_NOTE_CON	PTM0520	n/a	SIA77
S_NOTE_PROD_INT	PTM0520	n/a	SIA77
S_NOTE_ACCNT	PTM0520	n/a	SIA77
S_DOCK_TXN_LOG	PTM0010	All Paths	All Paths
S_ESCL_REQ	PTM0010	All Paths	All Paths

NOTE: Unload and load jobs on tables that contain CLOB data are processed by the PTMCLOBx macros. These macros adjust the unload and load job control cards to handle any CLOB data in a table. The PTMCLOBx macros can be found in the *DSNHLQ.SIEBEL.EXEC* or in the VSTG0300 staging data set.

PRET Members Generated By Pretedit.txt

The Pretedit.txt file creates the partitioned data set (PDS) on the z/OS host that is used for PRET processing. The members in this PDS perform a number of tasks, for example, listing the source tables that contain CLOB columns, gathering information required for key processing, and deleting rows in specific tables. Data sets are generated for each upgrade path.

Table 58 shows the data set members created by the Pretedit.txt file, the objects amended by these members (the SQL statement that is run is contained in the member) and the upgrade path for which these members are generated.

Table 58. PRET Processing Members Created By the Pretedit.txt File

PDS Member Name	PRET Object Affected	SIA Paths
PRETLDIN	S_DOCK_TXN_LOG	All
PRETLDIN	S_ESCL_REQ	All
PRETLDIN	S_SRM_REQUEST	SIA621, SIS63
PRETCLBF	SQL/CLOB list, used by SBLCLOBU	All
PRETKEYS	SQL/Clustering Index Key structures	All

Source Tables Amended During PRET Processing

The PRET upgrade jobs perform operations on the source tables listed in [Table 59](#). You might want to back up these tables before you start the upgrade.

[Table 59](#) shows the source tables amended by PRET upgrade processing, the type of amendment made (the table is either altered or dropped), and the upgrade path affected.

Table 59. Tables Changed During PRET Processing

Tables Amended During PRET Processing	Type of Change Made	SIA Paths
S_EMPLOYEE	Alter	SIA621, SIS63
S_INS_CLAIM	Alter	SIA621
S_ORG_GROUP	Alter	SIA621, SIS63
S_CON_ADDR	Alter	SIA704
S_FN_CRDT_RPT	Alter	SIA752
S_REGION	Alter	SIA704 (FINS704 only, n/a for SIS704), SIA752
S_EMP_TLR_LMTS	Drop	SIA621 only
S_FN_HLDVR_CTF	Drop	SIA621 only
S_FN_TXN_LMTS	Drop	SIA621 only

Index

Symbols

.srf file

See configuration file

Numerics

6.x

duplicate Row IDs, identifying and resolving 56, 63, 71, 78
repository, running Repository Preparation Wizard 277
Siebel Configuration data, preparing for upgrade 149

7.7, planning upgrade

new database schema structure 88

7.x, planning upgrade

backup and recovery stages, planning 93
capacity planning 91
code page support, considering 90
IBM utilities, obtaining required 92
Job Stream Optimization utility 393
new database schema structure in 7.7 88
schedule, creating 94
security privileges, obtaining required 92
Siebel-scheduled mode or vendor-schedule mode 91
software and hardware, required 92
testing before production upgrade 90

A

access control

about 98
S_CONTACT and S_ORG-GROUP, about multi-org visibility for 98

adding

Siebel user 132

aggregate categories (new),

reviewing 337

AIX memory allocation segment space,

securing 118, 145

applet functionality, reviewing 330

applets

7.x applet scripts, about migrating to 318
applet models, about and list of 305
customizations, migrating to applets and views 311
migrating applets to release 7.x 301

new applet migration, reviewing 314
scripts, about migrating to release 7.x 307, 318

Application scripts

7.x, about migrating to 307
migrating 321

attachments

file system attachments, updating 355

audience for guide 85

B

backing up

development database 131

backup and recovery stages, planning 93

business component definitions, changes to party model 101

Business Component scripts

7.x, about migrating to 307
migrating 321

business components

custom business component configurations, migrating 199

join conflicts, resolving 326

visibility properties, changing 364

Business Services scripts, migrating 321

C

Call Center, locating user logins that require resolution of duplicates 367

call report attachments

file names, updating 373, 384

updating (procedure) 373, 384

capacity planning 91

Catalog visibility

See visibility modes, setting for access control

client configuration, migrating to Web client

See also postupgrade tasks; preupgrade tasks
about 301

applet and view models 305

applet migration, reviewing 314

applet scripts, migrating to 7.x 318

applets and views, migrating, about and wizards 301

client-side interfaces, migration tasks 308

hidden properties, exposing 310

migrating Business Components, Business

- Services, and Application scripts 321
 - outbound COM interfaces, migrating 322
 - scripts, analyzing for migration 316
 - scripts, migration tasks 307
 - views, remigrating 315
- client-side interfaces**
 - outbound COM interfaces 308
 - outbound COM interfaces, migrating 322
- clustered indexes**
 - considerations for 102
- clustered indexes, considerations** 116
- code page**
 - considering support 90
- columns, denormalized**
 - 6.2.1 Siebel Business applications 430
 - 6.2.1 Siebel Financial Services applications 421
 - 6.3 Siebel Industry Solutions 438
 - 7.0.4 Siebel Business applications 448
 - 7.0.4 Siebel Financial Services applications 450
 - 7.0.5 Siebel Industry Solutions 454
 - 7.5.2 Siebel Financial Services applications 459
- configuration file**
 - new .srf file, compiling 297
 - new custom configuration file, producing 297
- copied objects**
 - upgrade inheritance, configuring objects for 270
- copied objects, automatic upgrade of**
 - upgrade ancestor, choosing 262
- custom business component configurations**
 - migrating 199
- custom database schema**
 - license keys, adding new 180
- custom indexes**
 - clustered indexes, considerations 116
 - custom temporary tables, dropping 116
- custom objects, remapping opportunity product migration objects** 375
- Customer Repository, preparing for merge**
 - See also* repository merge, performing
 - inheriting upgrade behavior, about 261
 - New Customer repository, preparing to import 295

D

- data migration**
 - party model, support of 100
- database schema**

- new structure in 7.7 88
 - testing before production upgrade 90
- database server**
 - resetting, about 349
- database server configuration, verifying** 117
 - See also* database sort order, verifying; database, preparing for upgrade
 - Siebel database expected growth during upgrade (table) 94
- database template file, regenerating** 348
- database, preparing for upgrade**
 - AIX memory allocation segment space, securing 118, 145
 - mobile and dedicated users, preparing for upgrade 140
 - New Customer Repository, preparing to import 295
- DB2, upgrade planning**
 - new database schema structure in 7.7 88
- dedicated users, preparing for upgrade** 140
- deleting**
 - duplicate EIM mappings 347
- denormalized columns**
 - See* columns, denormalized
- development environment upgrade**
 - integration objects, upgrading 363
- development environment, postupgrade tasks**
 - database server, about resetting 349
 - database template file, regenerating 348
 - developers or Siebel Tools clients, extracting 349
 - EIM mappings, deleting duplicate 347
 - obsolete tables, reapplying custom extensions 345
 - Siebel Configurator, postupgrade tasks 389
- development environment, preupgrade tasks**
 - virtual memory, about allocating sufficient 354
- development environment, upgrading**
 - 6.x duplicate Row IDs, identifying and resolving 56, 63, 71, 78
 - custom business component configurations, migrating 199
 - foreground performance, increasing 283
 - parameter validation, about 39
 - repositories for an ASCII database, merging 283
 - repositories for an DB2 EBCDIC database, merging 286

- repository merge, about 281
- repository merge, categories of object differences 279
- repository merge, optimizing performance 282
- Repository Preparation Wizard, running 277
- schema, synchronizing 198
- Siebel Upgrade Wizard, resuming after first pause 185
- Siebel Upgrade Wizard, resuming after second pause 193
- Siebel Upgrade Wizard, resuming after third pause 195
- SQL files, executing after first pause 183
- SQL output files, executing after second pause 186
- SQL output files, executing after third pause 195
- statistics, updating 197
- stored procedures and user-defined functions, deploying 188
- upgrade inheritance, configuring objects for 270
- development upgrade**
 - clustered indexes, considerations 116
 - custom temporary tables, dropping 116
 - database server configuration, verifying 117
 - DB2 software, upgrading 95
 - development database, backing up 131
 - development repository projects, checking in 152
 - EIM tables, preparing for upgrade 153
 - intersection table maintenance and deduplication, preparing for 155
 - prerequisites 151
 - repositories, preparing for upgrade 157
 - schema, preparing storage layout 119
 - Siebel Configurator data 6.x, preparing for upgrade 149
 - Siebel user, adding 132
 - storage control file, modifying 120
 - storage control file, reviewing 127
 - storage control file, validating 126
 - storage groups, creating 118
 - tablespace size, reviewing 129
- E**
- eChannel, about preparing for upgrade** 146
- EIM mapping**
 - deleting duplicates 347
- EIM tables**
 - preparing for upgrade 153
- Encryption Upgrade Utility**
 - RC2 encryption, about upgrading to 96
- Enterprise Integration Manager**
 - integration object, upgrading 363
- F**
- file system**
 - attachments, updating 355
- Forecasting data, preparing for upgrade** 146
- G**
- Generate Reporting Relationships button, exposing** 360
- global deployment, postupgrade tasks**
 - global time zone, about setting up environment to support 362
 - global time zone, enabling after an upgrade 362
 - UTC delta columns, upgrading 363
- global time zone**
 - See global deployment, postupgrade tasks
- grid-based applets, reviewing guide** 332
 - audience for 85
 - naming and typographical conventions 22
- H**
- hardware, obtaining and software** 92
- hidden properties, exposing** 310
- household notes, updating file system** 374
- I**
- IBM DB2 UDB, preparing database for upgrade**
 - clustered indexes, considerations for 102
- IBM utilities**
 - obtaining required 92
- ICL, about** 264
- importing**
 - development repository and schema definition 161
- Incorporate Custom Layouts, about indexes** 264
 - clustered indexes, considerations 116
 - custom temporary tables, dropping 116
- inheriting upgrade behavior, about** 261
- integration objects, upgrading** 363
- intersection tables**
 - preparing for maintenance and

deduplication 155
item identifier, Web template 343

J

Job Stream Optimization utility

using 393

join conflicts, resolving 326

L

license keys

custom database schema, adding new
 keys 180

log files

archiving manually 259
 SiebSrvr log files, about saving 180

M

Microsoft SQL Server

Siebel Software Configuration utility,
 changing language utility runs 163

migrating, client configuration to Web client

See also postupgrade tasks; preupgrade tasks
 about 301

applet and view models 305
 applet migration, reviewing 314
 applet scripts, migrating to 7.x 318
 applets and views, migrating 301
 client-side interfaces, migration tasks 308
 hidden properties, exposing 310
 migrating Business Components, Business
 Services, and Application
 scripts 321
 outbound COM interfaces, migrating 322
 scripts, analyzing for migration 316
 scripts, migration tasks 307
 views, remigrating 315

mobile users, preparing for upgrade 140

multilingual deployment, installing

additional languages 95

multilingual seed data

Siebel Database Server for DB2,
 installing 95

multi-value group shuttle applets,

reviewing 334

N

naming conventions 22

O

objects

opportunity product migration, remapping
 objects 375

upgrade inheritance, configuring objects
 for 270

outbound COM interfaces, migrating 322

P

party model

about 99
 business component definitions 101
 data migration, support of 100
 data model, changes to (diagram) 100
 obsolete tables in 7.x 100

picklist visibility properties, changing 365

planning

Employee Relationship Management (ERM),
 about 104

guidelines 85

handheld devices 111

Resonate Central Dispatch 112

Workflow Designer 110

postupgrade tasks

See also preupgrade tasks; RC2 encryption,
 upgrading to

asset-based ordering 381

business components and join conflicts,
 resolving 326

call report attachments, updating 373,
 384

dock object and rule definitions, verifying are
 correct 353, 373

file system attachments, updating 355

Generate Reporting Relationships button,
 exposing 360

household notes and attachments, updating
 file system 374

inbound workflows, updating 388

integration objects, upgrading 363

log files, manually archiving 259

new custom configuration file,
 producing 297

Purchase Orders 380

quotes 383

reporting relationships, generating
 (procedure) 360

reporting relationships, generating,
 about 360

seeded workflows, incorporating prior
 customizations into 388

seeded workflows, using 388

symbolic string model, upgrading to 349

Transaction Amount field in Payment lines,
 updating 380

Universal Inbox 379

user logins, locating that require resolution of

- duplicates 367
 - visibility modes, setting for access control 364
 - preupgrade tasks**
 - AIX memory allocation segment space, securing 118, 145
 - mobile and dedicated users, preparing for upgrade 140
 - New Customer Repository, preparing to import 295
 - procedures**
 - stored procedures, deploying and user-defined functions 188
 - production environment repositories**
 - preparing for upgrade 162
 - production environment upgrade**
 - development environment, upgrading without 180
 - global time zone, about operating in 362
 - New Customer Repository, preparing to import 295
 - preparing production upgrade on UNIX 169
 - Transaction Amount field in Payment Lines, updating 380
 - production environment, upgrading**
 - channel data, migrating 233
 - configuration data, migrating 235
 - control cards and schema DDL files, transferring to zSeries 215
 - FINS, migrating configuration data 236
 - Gen Primary 7.7 indexes. rebuilding 246
 - Gen Primary 7.7 SQL, migrating 246
 - household data, migrating 234
 - index DDL, executing and rebuild indexes 230
 - index schema, transferring to zSeries 239
 - interface tables, generating and dropping 212
 - intersection tables, maintaining 249
 - ISS data, migrating 237
 - JCL, preparing 240
 - JCL, SQL, and binary files, transferring to zSeries 204
 - load jobs, running on the target database 229
 - log tables, loading on the source database 226
 - log tables, loading on the target database 227
 - new EIM indexes, creating 247
 - old indexes, dropping and creating new indexes 244
 - preschm data, migrating 231
 - preschm data, migrating for FINS 234
 - schema and JCL files, preparing on zSeries 216
 - Siebel Upgrade Wizard, resuming after first pause 213
 - Siebel Upgrade Wizard, resuming after second pause 238, 251
 - Siebel-Scheduled jobs, preparing on zSeries 221
 - source database job stream, optimizing 224, 242
 - source database, running unload jobs 228
 - stored procedure, installing on zSeries 219
 - table creating maintenance jobs, running 213
 - table edit maintenance jobs, running 212
 - target Siebel log tables, loading 244
 - zSeries setup dataset, defining and allocating 205
 - zSeries upgrade environment, setting up 207
 - zSeries, prerequisites for upgrade 203
 - production environments, postupgrade tasks**
 - mobile and connected clients, upgrading 351
 - regional servers, upgrading 351
 - production upgrade**
 - testing before 90
 - production upgrade, prerequisites**
 - about 159
 - development repository and schema definition, importing 161
 - production environment repositories, preparing for upgrade 162
 - source database values 160
 - target database values 160
- Q**
- quotes**
 - upgrade, preparing for 147
 - Quotes data, preparing for upgrade** 147
- R**
- RC2 encryption, upgrading to**
 - about 96
 - recovery**
 - planning and backup stages 93
 - reporting relationships**
 - about generating 360
 - Generate Reporting Relationships button, exposing 360
 - generating (procedure) 360
 - repositories**

- development environment repositories, preparing for upgrade 157
- production environment repositories, preparing for upgrade 162
- repositories for an ASCII database, merging 283
- repositories for an DB2 EBCDIC database, merging 286
- repository merge, about 281
- repository merge, categories of object differences 279
- repository merge, increasing the foreground performance 283
- repository merge, optimizing performance 282
- repositories, upgrading**
 - about 49
 - EIM temporary columns, generating 294
 - Incorporate Custom Layouts (ICL) 264
 - postmerge utilities, about 264
 - upgrade behavior, inheriting 261
- repository merge results**
 - determining success 288
 - reviewing conflicts 291
 - reviewing deleted objects 292
 - reviewing obsolete objects 293
- Repository Preparation Wizard**
 - running on 6.x repository 277
- rich text controls, revising** 336
- Row IDs**
 - 6.x, identifying and resolving 56, 63, 71, 78
- RTCs, revising** 336

- S**
- S_CONTACT, multi-org visibility for** 98
- S_LIT files, upgrading to S_CB_ASSET_VER files** 355
- S_ORG_GROUP, multi-org visibility for S_PARTY** 98
 - See party model
- S_SRC_PAYMENT table, about columns added** 101, 150
- schedule**
 - planning version 7.x 94
- schema**
 - storage layout, preparing 119
 - synchronizing 198
- scripts**
 - applet scripts, about migrating to 7.x 318
 - business components, business services, and applications scripts, migrating 321
 - migrating to release 7.x 307
 - migration, analyzing for 316
- security**
 - obtaining required privileges 92
- seed data, installing multilingual seed data** 95
- shuttle applets, reviewing** 334
- Siebel Application Integration (EAI), about using SAP and upgrading** 25
- Siebel Business application, upgrading**
 - multilingual deployment, installing additional languages 95
- Siebel Business applications**
 - denormalized columns for 6.2.1 430
 - denormalized columns for 7.0.4 448
- Siebel Configurator**
 - 6.x data, preparing for upgrade 149
 - postupgrade tasks 389
- Siebel data model**
 - access control, about 98
 - additional considerations 101
 - party model, business component definitions 101
 - party model, changes to data model (diagram) 100
 - party model, data migration in support of 100
 - party table, about 99
 - party table, obsolete tables in 7.x 100
 - S_SRC_PAYMENT Table, columns added to 101, 150
- Siebel Database Schema, upgrading**
 - restarting upgrade 170
- Siebel Database Server software**
 - multilingual deployment, installing additional languages 95
- Siebel eChannel, about preparing for upgrade** 146
- Siebel Enterprise Integration Manager**
 - deleting duplicate EIM mappings 347
- Siebel Financial Services**
 - application tables, preparing for upgrade 149
 - call report attachments, updating 373, 384
 - call report attachments, updating file names 373, 384
 - denormalized columns for 6.2.1 421
 - denormalized columns for 7.0.4 450
 - denormalized columns for 7.5.2 459
 - household data integrity, verifying 147
 - household notes and attachments, updating file system 374
 - opportunity product migration 375
- Siebel Industry Solutions**

- denormalized columns for 6.3 438
- denormalized columns for 7.0.4 454
- Siebel Marketing**
 - obsolete 7.x business objects 407
 - obsolete 7.x data 401
 - obsolete 7.x views 408
- Siebel Purchase Orders, updating**
 - Transaction Amount field** 380
- Siebel Quotes, about preparing upgrade for** 147
- Siebel Repository, preparing for upgrade**
 - dock objects and visibility rules, recording 117
- Siebel Software Configuration utility**
 - language, changing 163
 - The Siebel Upgrade Wizard, about 40
- Siebel Upgrade Wizard**
 - about 40
 - first pause, resuming after 185
 - second pause, resuming 193
 - third pause, resuming after 195
- Siebel user**
 - adding 132
- Siebel Workflow**
 - inbound workflow, updating 388
 - seeded workflows, incorporating prior customizations into 388
 - seeded workflows, using 388
- SiebSrvr log files, about saving** 180
- software, obtaining and hardware SQL** 92
 - output files, about executing after second pause 186
 - output files, about executing after second pause stage 2 193
 - output files, executing after second pause stage 1 187
 - output files, executing after third pause 195
- .srf file**
 - See configuration file
- statistics, updating** 197
- storage control file**
 - modifying 120
 - reviewing 127
 - validating 126
- storage groups**
 - creating 118
- stored procedures**
 - deploying and user-defined functions 188

T**tables**

- custom temporary tables, dropping 116
- EIM tables, preparing for upgrade 153
- intersection tables, preparing maintenance and deduplication 155
- obsolete tables, reapplying custom extensions 345
- table creation maintenance jobs, running 213
- table edit maintenance jobs, running 212
- tables, preparing for upgrade**
 - clustered indexes, considerations 102
 - DB2 UDB long columns, identifying for truncation 94, 116, 117, 140, 295
- tablespace**
 - size, reviewing 129
- testing**
 - before production upgrade 90
- Transaction Amount field in Payment lines, updating** 380
- typographical conventions** 22

U

- upgrade inheritance**
 - configuring objects for 270
- Upgrade Planning Worksheet** 417
- upgrade, planning for**
 - resources 83, 391, 392, 393
- upgrading ancestor, choosing an** 262
- upgrading inheritance**
 - parent to descendents, propagating changes from 263
- user**
 - adding Siebel user 132
- UTC conversion utility, upgrading** 363
- utilities**
 - IBM utilities, obtaining required 92
 - Job Stream Optimization utility, using 393

V

- version 6.x**
 - duplicate Row IDs, identifying and resolving 56, 63, 71, 78
 - repository, running Repository Preparation Wizard 277
 - Siebel Configurator data, preparing for upgrade 149
- version 7.7, planning upgrade**
 - new database schema structure 88
- version 7.x, planning upgrade**
 - backup and recovery stages, planning 93
 - capacity planning 91
 - code page support, considering 90

- IBM utilities, obtaining required 92
- Job Stream Optimization utility 393
- new database schema structure in 7.7 88
- schedule, creating 94
- security privileges, obtaining required 92
- Siebel-scheduled mode or vendor-schedule mode 91
- software and hardware, required 92
- testing before production upgrade 90
- view tabs, verifying display** 340
- views**
 - customization, migrating to applets and views 311
 - migrating to release 7.x 301
 - remigrating views 315
 - view models, about and list of 305
 - visibility properties, changing 366
- virtual memory**
 - development environment, about allocating sufficient 354
- visibility filters, reviewing** 338
- visibility modes, setting for access control**
 - about and Catalog visibility 364
 - Auto Query Mode, picklists that start in (list of) 365
 - business component visibility properties, changing 364
 - business components, visibility type used (list of) 364
 - picklists visibility properties, changing 365
 - views visibility properties, changing 366
 - views, visibility type used (list of) 366

W

- Web Client Migration Wizard**
 - customizations, migrating to applets and views 311
- Web client scripting, about in release 7.x** 307
- Web template item identifier** 343
- workflows**
 - inbound, updating 388
 - seeded, incorporating prior customizations into 388
 - seeded, using 388
 - test to production, transitioning 389
- worksheet**
 - Upgrade Planning 417

Z

- zSeries**
 - control cards and schema DDL files, transferring 215
 - index schema, transferring to 239
 - JCL, SQL, and binary files, transferring to zSeries 204
 - schema and JCL files, preparing 216
 - setup dataset, defining and allocating 205
 - Siebel-Scheduled jobs, preparing on 221
 - stored procedures, installing on 219
 - upgrade environment, setting up 207
 - upgrade prerequisites 203