

---

# PeopleSoft Enterprise Campus Solutions 8 SP1 to HRMS and Campus Solutions 9.0 Upgrade

---

**February 2007**

PeopleSoft Enterprise Campus Solutions 8 SP1 to HRMS and Campus Solutions 9.0  
Upgrade  
SKU uhrfo9\_021507\_ucs801to90  
Copyright © 2007, Oracle. All rights reserved.

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

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

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

### **U.S. GOVERNMENT RIGHTS**

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

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

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

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

### **Open Source Disclosure**

Oracle takes no responsibility for its use or distribution of any open source or shareware software or documentation and disclaims any and all liability or damages resulting from use of said software or documentation. The following open source software may be used in Oracle’s PeopleSoft products and the following disclaimers are provided.

#### *Apache Software Foundation*

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright © 2000-2003. The Apache Software Foundation. All rights reserved. Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>.

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

### *OpenSSL*

Copyright © 1998-2005 The OpenSSL Project. All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### *Loki Library*

Copyright © 2001 by Andrei Alexandrescu. This code accompanies the book: Alexandrescu, Andrei. “Modern C++ Design: Generic Programming and Design Patterns Applied”. Copyright © 2001 Addison-Wesley. Permission to use, copy, modify, distribute and sell this software for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

### *Helma Project*

Copyright © 1999-2004 Helma Project. All rights reserved. THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HELMA PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Helma includes third party software released under different specific license terms. See the licenses directory in the Helma distribution for a list of these license.

### *Sarissa*

Copyright © 2004 Manos Batsis.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

### *ICU*

ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright © 1995-2003 International Business Machines Corporation and others. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

*Sun's JAXB Implementation – JSDK 1.5 relaxngDatatype.jar 1.0 License*

Copyright © 2001, Thai Open Source Software Center Ltd, Sun Microsystems. All rights reserved.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*W3C IPR SOFTWARE NOTICE*

Copyright © 2000 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

Note: The original version of the W3C Software Copyright Notice and License could be found at <http://www.w3.org/Consortium/Legal/copyright-software-19980720>.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

# Contents

## Preface

<b>About This Documentation.....</b>	<b>xxvii</b>
Understanding This Documentation.....	xxvii
Prerequisites.....	xxvii
Audience.....	xxvii
Organization.....	xxviii
Related Information.....	xxviii

## Chapter 1

<b>Prepare Your Database.....</b>	<b>1</b>
Understanding Database Preparation.....	1
Understanding Your Upgrade.....	2
Understanding PeopleSoft Upgrades.....	2
Verifying the Software Installation.....	2
Defining Upgrade Databases.....	2
Reviewing Upgrade Notes and Tips.....	3
Copying Your Production Database .....	4
Making a Copy of Production Database.....	5
Increasing Database Space.....	5
Applying Upgrade Planning Files.....	5
Understanding Applying Upgrade Planning Files.....	6
Applying the UPGOPT Project.....	6
Building the UPGOPT Project.....	6
Setting Up Upgrade Planning Security.....	7
Editing Upgrade Planning DB2 Scripts.....	8
Running Upgrade Planning Scripts.....	8
Understanding Running Upgrade Planning Scripts.....	9
Loading Planning Messages and Strings.....	9
Populating Recruiting Solutions .....	9
Populating the Equation Renames .....	9
Importing Upgrade Default Options.....	9
Updating Statistics.....	10
Running Initial Audit Reports.....	10
Understanding Running Initial Audit Reports.....	10

Run the Initial DDDAUDIT Report.....	10
Run the Initial SYSAUDIT Report.....	11
Run the Initial Alter Audit.....	11
Review the Initial Audits.....	15
Running Application Audits.....	16
Running Upgrade Reports.....	16
Correcting Garnishment Payee Data.....	21
Preparing for Personal Data Changes.....	22
Running Dynamic ChartFields Audits.....	24
Preparing Approvals for Upgrade.....	25
Understanding Approvals Upgrade.....	26
Completing eProfile Approvals.....	26
Preparing the Benefits Upgrade.....	26
Preparing Benefits Administration.....	26
Preparing COBRA Administration.....	27
Making Functional Decisions.....	27
Understanding Making Functional Decisions.....	27
Assigning Upgrade Default Values.....	28
Assigning Training Employee Record Default Values.....	39
Preparing Workflow Transactions.....	40
Completing eCompensation Transactions.....	40
Completing eDevelopment Transactions.....	41
Mapping Equation Values.....	41
Exporting Upgrade Default Options.....	43
Upgrading Defaults for Service Indicators.....	44
Reporting Row Count for Tables.....	48
Preparing Your Database .....	48
Understanding Database Preparation.....	49
Verify Database Integrity.....	49
Clean the PSOBJCHNG Table.....	49
Shrink Images.....	49
Purge Message Queues.....	50
Backing Up Demo Databases.....	51
Backing Up Copy of Current Demo.....	51
Backing Up New Release Demo.....	51
Renaming Records and Fields .....	51
Understanding Renaming Records and Fields.....	52
Running the RNHCUPG01 Script.....	52
Running the RNHCEOG20 Script.....	52
Running the RNCSUPG03 Script.....	53

Retaining the Target Rename Log Files.....	53
Running RNHCUPG01 Script on Copy of Current Demo.....	53
Running RNHCEOG20 Script on Copy of Current Demo.....	54
Running RNCSUPG03 Script on Copy of Current Demo.....	54
Comparing Customizations.....	54
Understanding Compare UPGCUST.....	54
Running Compare UPGCUST.....	55
Running UPGCUST Filter Script.....	56
Reviewing UPGCUST Compare Log.....	56
Restoring Copy of Current Demo.....	56
Identifying Customizations.....	57
Preparing for the Application Upgrade.....	59
Running the XLATTABLE Copy Script.....	59
Creating a Copy of RecField Definitions.....	59
Dropping Student Financials Tables.....	60
Copying Time and Labor Temp Table List.....	60
Reversing Renames for Microsoft SQL Server.....	61
Renaming Table with Data Type Changes for Microsoft.....	61
Renaming DI Tables for Microsoft.....	61
Renaming DI Tables for Oracle.....	62
Renaming DI Tables for Informix.....	62
Renaming DI Tables for Sybase.....	63
Renaming DI Tables for DB2 zOS.....	63
Renaming DI Tables for DB2 UNIX NT.....	63
Editing the System Data Swap Script.....	64
Editing the PeopleTools Swap Script.....	64
Backing Up After Prepare Your Database.....	65

## Chapter 2

<b>Apply PeopleTools Changes.....</b>	<b>67</b>
Understanding PeopleTools Changes.....	68
Verifying Upgrade User.....	68
Performing Script Modifications.....	69
Understanding Script Modifications.....	69
Copy the PTDDLUPG Script.....	70
Edit the PTDDLUPG Script.....	70
Run a DBTSFIX Report.....	71
Edit the DBTSFIX Output Scripts.....	71
Edit the GRANT Script.....	72

Edit the RELGA Script.....	72
Edit the PSLANG Script.....	72
Edit the PSLANGUAGES Script.....	73
Edit the TLSUPGNONCOMP Script.....	73
Edit the PTxxxTLS Scripts.....	73
Edit the DB2 Scripts.....	74
Edit Move to Production Import Scripts.....	75
Edit Move to Production Password.....	76
Edit the DDLDB2 Script.....	76
Edit the DDLDBX Script.....	76
Edit the DDLORA Script.....	77
Edit the DDLIFX Script.....	77
Edit the MSGTLSUPG Script.....	77
Edit the Integration Broker Script.....	78
Edit Multilingual Step Properties.....	78
Edit the Message Nodes Script.....	79
Edit Data Type Steps.....	80
Updating PeopleTools System Tables .....	80
Understanding Updating PeopleTools System Tables.....	81
Clean Up Message Data.....	81
Create Tablespaces.....	82
Create Tablespaces for Informix.....	82
Update System Catalog Views.....	82
Update PeopleTools System Tables.....	83
Grant Privileges to the CONNECT ID.....	83
Run the RELGA Script.....	84
Populate Languages for Login.....	84
Update the Product License Code.....	84
Update PeopleTools Patch Information.....	85
Create Temporary Performance Indexes.....	86
Export PeopleTools System Tables.....	86
Import PeopleTools System Tables.....	86
Reset Database Options Flag.....	87
Rerun Update Statistics for DB2 zOS.....	87
Rerun RUNSTATS Report for DB2 UNIX NT.....	87
Rerun Update Statistics for DB2 UNIX NT.....	88
Rerun Update Statistics for Informix.....	88
Rerun Update Statistics for Oracle .....	88
Turning Off Change Control .....	89
Loading Model Definition Data.....	89



Understanding Model Definition Data Load Process.....	89
Load Model Definitions for DB2 zOS.....	89
Load Model Definitions for DB2 UNIX NT.....	90
Load Model Definitions for Oracle.....	90
Load Model Definitions for Informix.....	90
Load Model Definitions for Microsoft.....	90
Load Model Definitions for Sybase.....	91
Loading Message Data.....	91
Renaming PeopleTools Records.....	91
Comparing PeopleTools Objects .....	92
Understanding Comparing PeopleTools Objects.....	92
Compare Fields in the PPLTOOLS Project.....	92
Review the Field Compare Report.....	94
Exporting and Copying Projects .....	94
Understanding Exporting and Copying Projects.....	94
Export and Copy PTPORTALDEL Project.....	95
Export and Copy PPLTOOLS Project.....	95
Export and Copy PPLDELETE Project.....	96
Export and Copy PPLTLS84CUR Project.....	96
Export and Copy PPLTLS84CURML Project.....	97
Export and Copy PPLTLSML Project.....	97
Export and Copy PPLTLS84CURDEL Project.....	98
Export and Copy PATCH84X Project.....	98
Export and Copy PATCH84XML Project.....	99
Populating Tablespace Data.....	99
Create Application Tablespaces.....	100
Populate Tablespace Data.....	100
Update Tablespace Names.....	101
Building the PeopleTools Project.....	102
Generate the PeopleTools Script.....	102
Edit the PeopleTools Script.....	102
Run the PeopleTools Script.....	103
Building Updated PeopleTools Project.....	103
Generate Updated PeopleTools Script.....	103
Edit the Updated PeopleTools Script.....	104
Run the Updated PeopleTools Script.....	104
Migrating Records to New Tablespaces.....	104
Understanding Migration of Records to New Tablespaces.....	105
Export and Copy the PT84TBLSPC Project.....	105
Build the Tablespace Alter Script.....	105

Edit the Tablespace Alter Script.....	105
Run the Tablespace Alter Script.....	106
Loading Base Data.....	106
Loading Language Data.....	107
Populate Languages.....	107
Load Language Data.....	107
Loading PeopleTools Data.....	107
Load NonComparable Objects.....	108
Load English Messages.....	108
Load English String Data.....	108
Load Stored Statements Data.....	108
Creating PeopleTools Views.....	109
Create PPLTOOLS Views.....	109
Create Updated PeopleTools Views.....	109
Converting PeopleTools Objects .....	109
Expand Fields on Records.....	110
Alter Approval Rule Set Related Tables.....	110
Update Message Nodes.....	110
Update Message Nodes for Informix.....	111
Convert Portal Objects.....	111
Convert Query Prompt Headings.....	112
Load Conversion Data.....	113
Report Conversion Details.....	113
Run Data Conversion.....	113
Converting Integration Broker.....	114
Updating Integration Broker Defaults.....	114
Creating Integration Broker Objects.....	114
Saving Application Messaging Objects.....	115
Exporting Node Transactions.....	115
Preparing Integration Broker Deletes.....	115
Deleting Application Messaging Objects.....	116
Deleting Node Transactions.....	116
Verifying NonBase Language Pages.....	116
Understanding NonBase Language Pages.....	116
Verify NonBase Language Pages.....	116
Delete Unused Language Pages.....	118
Clearing Process and Report Tables.....	118
Understanding Process and Report Tables.....	119
Purge Process Request Tables.....	119
Setting Object Version Numbers.....	119

Converting Database Data Types.....	119
Understanding Converting Database Data Types.....	121
Backing Up Before Platform Changes.....	121
Running Long Data Audit.....	122
Validating Microsoft Database.....	122
Reviewing Microsoft Settings.....	122
Creating Microsoft Conversion Project.....	123
Generating Microsoft Conversion Script.....	123
Running Microsoft Conversion Script.....	123
Granting Permissions to the CONNECT ID.....	123
Running Microsoft Conversion Report.....	124
Validating Oracle Database.....	124
Creating Oracle Audit Tables.....	124
Auditing Duplicate Length Constraints.....	125
Auditing Disabled Constraints.....	125
Reviewing Oracle Settings.....	125
Generating Oracle Conversion Scripts.....	126
Running Long to LOB Script 1.....	128
Running Long to LOB Script 2.....	128
Running Long to LOB Script 3.....	128
Running Long to LOB Script 4.....	128
Running Long to LOB Script 5.....	129
Running Long to LOB Script 6.....	129
Running Long to LOB Script 7.....	129
Running Long to LOB Script 8.....	129
Auditing Long to LOB Conversion.....	130
Running Character Length Script 1.....	130
Running Character Length Script 2.....	130
Running Character Length Script 3.....	130
Running Character Length Script 4.....	131
Running Character Length Script 5.....	131
Running Character Length Script 6.....	131
Running Character Length Script 7.....	131
Running Character Length Script 8.....	132
Auditing Character Length Semantics.....	132
Reviewing Conversion Reports.....	132
Updating Database Options.....	133
Backing Up After PeopleTools Upgrade.....	133

**Chapter 3**

<b>Run and Review Compare Reports.....</b>	<b>135</b>
Understanding Compare Reports.....	135
Preparing for Application Changes.....	135
Running Drop Table Script.....	135
Loading HR Ethnicities Table.....	136
Exporting Project Definitions.....	136
Importing Project Definitions.....	136
Dropping View ADP_ACCT_CD_VW.....	137
Running the Personal Data Setup Script.....	137
Renaming Fields for Microsoft SQL Server.....	138
Renaming Tables.....	138
Understanding Renamed Tables.....	139
Running the RNHCUPG03MSS Script.....	139
Running the RNHCUPG03DB2 Script.....	139
Running the RNHCUPG03DBX Script.....	139
Running the RNHCUPG03IFX Script.....	140
Running the RNHCUPG03ORA Script.....	140
Running the RNHCUPG03SYB Script.....	140
Running the RNCSUPR04MSS Script.....	140
Running the RNCSUPR04DB2 Script.....	141
Running the RNCSUPR04DBX Script.....	141
Running the RNCSUPR04IFX Script.....	141
Running the RNCSUPR04ORA Script.....	141
Running the RNCSUPR04SYB Script.....	142
Running the RNCSUPG02MSS Script.....	142
Running the RNCSUPG02DB2 Script.....	142
Running the RNCSUPG02DBX Script.....	142
Running the RNCSUPG02IFX Script.....	143
Running the RNCSUPG02ORA Script.....	143
Running the RNCSUPG02SYB Script.....	143
Running New Release Compare Reports.....	143
Understanding the New Release Compare.....	144
Preserving Local Message Node.....	144
Modifying New Release Compare Options.....	144
Running New Release UPGCUST Compare.....	146
Running New Release UPGCUSTIB Compare.....	146
Resetting Take Action Flags in UPGCUST.....	146
Running UPGCUSTIB Filter Script.....	147
Reviewing New Release Compare Reports.....	147

Reviewing New Release Changes.....	147
Reviewing NonComparable Objects.....	148

## Chapter 4

<b>Apply Application Changes.....</b>	<b>149</b>
Understanding Application Changes.....	149
Customizing the New Release.....	150
Understanding New Release Customizations.....	150
Copying the UPGCUST Project.....	150
Reviewing Copy Results.....	150
Running New Release Upgrade Copy.....	151
Exporting Selected PeopleTools Tables.....	151
Importing Selected PeopleTools Tables.....	152
Copying the UPGCUSTIB Project.....	152
Swapping PeopleTools Tables.....	152
Creating the UPGIBCOPY Project.....	153
Copying the UPGIBCOPY Project.....	154
Updating Target Values.....	154
Copying the UPGNONCOMP Project.....	154
Reviewing Project Copy Results.....	155
Exporting New Release Objects.....	155
Importing New Release Objects.....	155
Resetting Object Version Numbers.....	155
Backing Up New Release Demo Again.....	156
Updating Database Overrides.....	156
Understanding Database Overrides.....	156
Set Index Parameters After Copy .....	156
Set Tablespace Names After Copy.....	157
Set Record Parameters After Copy.....	157
Create New Tablespaces.....	158
Backing Up After the Upgrade Copy.....	160
Modifying the Database Structure.....	160
Understanding Modifying the Database Structure.....	161
Backing Up for DB2.....	161
Building the Upgrade Tables Script.....	162
ReCreating Upgrade Tables.....	162
Building Renamed Tables Script.....	162
Running the Renamed Tables Script.....	163
Creating the ALLTEMPTABS Project.....	163

Building the Create Temp Tables Script.....	163
Creating the ALLTABS Project.....	164
Building the Create and Alter Scripts.....	164
Editing the Create and Alter Scripts.....	165
Creating Temporary Tables.....	165
Creating Tables.....	166
Altering Tables.....	166
Creating Indexes.....	166
ReCreating Triggers.....	167
Reviewing the Create Indexes Log.....	167
Dropping Indexes for Data Conversion.....	167
Creating Indexes for Data Conversion.....	168
Setting Index Parameters.....	171
Setting Temp Table Tablespace Names.....	171
Setting Tablespace Names.....	171
Setting Record Parameters.....	172
Generating a DB2 UNIX RUNSTATS Script .....	172
Updating Statistics for DB2 UNIX.....	172
Updating Statistics for DB2 zOS.....	173
Updating Statistics for Informix.....	173
Updating Statistics for Oracle.....	173
Loading Data for Data Conversion.....	173
Swapping Languages on System Data.....	174
Exporting Application Messages.....	175
Importing Application Messages.....	175
Exporting Record Groups.....	175
Importing Record Groups.....	175
Exporting the System Setup Data .....	176
Importing the System Setup Data.....	177
Exporting the PW Pagelet Data.....	177
Importing the PW Pagelet Data.....	177
Exporting the Pagelet Wizard Data.....	178
Importing the Pagelet Wizard Data.....	178
Exporting Upgrade Defaults.....	178
Importing Upgrade Defaults.....	179
Exporting Delivered Equations.....	179
Importing Delivered Equations.....	179
Exporting Application Conversion Data.....	180
Importing Application Conversion Data.....	180
Exporting Data Conversion Driver Data.....	180

Importing Data Conversion Driver Data.....	180
Populating Control Tables for Data Conversion.....	181
Restoring New Release Demo.....	181
Applying Updates Before Data Conversion.....	181
Configuring Scheduler and Server.....	182
Backing Up Before Data Conversion.....	183
Running Data Conversion .....	183
Reviewing Data Conversion Tips.....	184
Running Data Conversion for Group 1.....	186
Running Data Conversion for Group 2.....	186
Running Data Conversion for Group 3.....	186
Running Data Conversion for Group 4.....	187
Running Data Conversion for Group 6.....	187
Running Data Conversion for Group 7.....	187
Running Data Conversion for Group 8.....	187
Running Data Conversion for Group 10.....	188
Running Data Conversion for Group 11.....	188
Backing Up After Data Conversion.....	188
Finalizing the Database Structure.....	188
Understanding Final Database Structure.....	189
Building the Alter with Deletes Scripts.....	189
Altering Tables with Deletes.....	189
Creating Indexes Again.....	190
Creating Triggers.....	190
Running the AE SYNCIDGEN Process.....	190
Creating All Views.....	190
Loading Data to Complete System Setup.....	191
Exporting Strings.....	192
Importing Strings.....	192
Exporting EDI Statements.....	192
Importing EDI Statements.....	192
Exporting Mass Change Data.....	193
Importing Mass Change Data.....	193
Exporting XML Service Information.....	193
Importing XML Service Information.....	194
Exporting Related Language System Data.....	194
Importing Related Language System Data.....	194
Editing the System Data Scripts.....	195
Exporting Application System Data.....	195
Importing Application System Data.....	195

Exporting Data for Your Upgrade Path.....	196
Importing Data for Your Upgrade Path.....	196
Exporting Common Portal System Options.....	196
Importing Common Portal System Options.....	197
Running the DLHCCOG03E Script.....	197
Running the DLHCCOG03I Script.....	197
Exporting Setup Data.....	197
Importing Setup Data.....	198
Exporting Approval Framework Definitions.....	198
Importing Approval Framework Definitions.....	198
Exporting Generic Notifications.....	198
Importing Generic Notifications.....	199
Loading Stored Statements.....	199
Understanding Loading Stored Statements.....	200
Setting PSOPTIONS for COBOL.....	200
Running the STOREFA Script.....	200
Running the STORECC Script.....	200
Running the STOREAD Script.....	201
Running the STOREAV Script.....	201
Running the STORESF Script.....	201
Running the STORESR Script.....	201
Running the STOREBAS Script.....	202
Running the STOREGP Script.....	202
Running the STOREHRM Script.....	202
Running the STOREPAY Script.....	202
Running the STOREPEN Script.....	203
Running the STOREPYI Script.....	203
Running Final Update Statistics.....	203
Generating Final RUNSTATS for DB2 UNIX.....	203
Running Final Statistics for DB2 UNIX.....	204
Running Final Statistics for DB2 zOS.....	204
Running Final Statistics for Informix.....	204
Running Final Statistics for Oracle.....	205
Completing Application Processes.....	205
Refreshing Personal Data Table.....	205
Running Data Conversion for Group 20.....	205
Exporting Setup Data for Rules.....	206
Importing Setup Data for Rules.....	206
Exporting Payroll Interface Tables.....	206
Importing Payroll Interface Tables.....	207



Exporting GL Interface Setup Tables.....	207
Importing GL Interface Setup Tables.....	208
Exporting US Custom Garnishment Rules.....	208
Importing US Custom Garnishment Rules.....	209
Updating Language Data.....	209
Understanding Updating Language Data.....	209
Run the TSRECPOP Script.....	209
Updating Object Version Numbers.....	210
Running the Final Audit Reports.....	210
Run the Final DDDAUDIT Report.....	210
Run the Final SYSAUDIT Report.....	211
Create the FNLALTAUD Project.....	211
Run the Final Alter Audit.....	211
Review the Final Audits.....	212
Run a Final SETINDEX Report.....	212
Run a Final SETTABLE Report.....	213

## Chapter 5

<b>Complete Database Changes.....</b>	<b>215</b>
Understanding Database Changes.....	216
Configuring the Upgrade Environment.....	216
Configure Application Server.....	216
Configure Portal.....	216
Reapplying Customizations.....	217
Understanding the Reapplication.....	217
Performing Customized Object Adjustment.....	218
Registering Portal Navigation Objects.....	218
Setting Up Security.....	219
Understanding Security.....	219
Set Up Security.....	219
Synchronize CREF Permissions.....	220
Grant Access to Personalize Homepage.....	221
Updating Department Security.....	222
Understanding Department Security.....	222
Refreshing Operator Security.....	222
Refreshing Transaction Records.....	223
Backing Up Before Manual Changes.....	223
Converting Retirees with Pay.....	224
Understanding the Conversion for Retirees with Pay.....	224

Running Retiree Additional Pay Report UVPYT05.....	227
Running Retiree Additional Pay Report UVPYT06.....	227
Running Retiree Additional Pay Report UVPYT07.....	228
Running Upgrade Conversion for Retirees with Pay.....	228
Rebuilding Job Flags.....	229
Verifying Equation Conversion.....	230
Understanding the Equation Conversion.....	230
Running the Equation Conversion Reports.....	230
Reviewing Manual Conversion Candidates.....	231
Reviewing Equations.....	231
Reviewing Equation Security.....	232
Running the ECPrep Process.....	233
Running the Person Organization Audits.....	234
Understanding the Person Organization Audits.....	234
Running Conversion Audits.....	235
Setting Up Pension.....	235
Reviewing the Benefits Limit Table.....	236
Setting Up Stock Administration .....	236
Understanding Stock Administration.....	237
Setting Up the Stock Table.....	237
Setting Up the Stock Purchase Plan Rules.....	237
Validating Alternative Overtime Upgrade.....	238
Updating Payroll Interface Definitions.....	238
Understanding Updates to Payroll Interface Definitions.....	239
Running Validate PI Field References Report.....	239
Updating PS Table Definitions.....	240
Updating Field Definition Table.....	240
Updating Instance Table Definitions.....	241
Upgrading Rules.....	241
Understanding Rules Upgrade.....	241
Running the Dropped Temp Tables Report.....	241
Recompiling Template Built Rules.....	242
Customizing Template Built Rules.....	245
Reviewing and Modifying User Exit Rules.....	247
Modifying SQL Statements for PROCESS_INSTANCE.....	248
Deleting Resume Files.....	249
Setting Up General Ledger Interface.....	249
Understanding GL Interface Setup.....	249
Setting Up the Liability Accounts.....	250
Setting Up the GL Activity Groupings.....	252

Setting Up the Expenses ChartField Mappings.....	252
Validating EE Garn Payee Data.....	253
Understanding EE Garn Payee Data Validation.....	253
Running the EE Garn Payee Data Validation Report.....	253
Updating Garn Payee Data.....	255
Setting Up US Custom Garnishment Rules.....	257
Understanding US Custom Garnishment Rules Setup.....	257
Creating US Custom Garnishment Rules.....	258
Cloning US Custom Garnishment Rules.....	258
Validating EE Garn Rule Data.....	259
Understanding EE Garn Rule Data Validation.....	259
Running the EE Garn Rule Data Validation Report.....	260
Updating Garn Rule Data.....	261
Updating the Employee ID Length.....	264
Setting Up Student Group Security.....	265
Reviewing PeopleTools Functionality.....	265
Preparing the Content Provider Registry.....	266
Updating the Portal Options Data.....	267
Stamping the Database.....	267
Reviewing Change Control.....	268
Backing Up Before Testing.....	269
Testing Your Copy of Production.....	269

## Chapter 6

<b>Apply Changes to Production Database.....</b>	<b>271</b>
Understanding the Move to Production.....	271
Testing the Move to Production.....	271
Understanding the Move to Production Test.....	271
Understand Move to Production.....	272
Creating a New Change Assistant Job.....	273
Testing Once More.....	274
Performing the Move to Production.....	274
Completing the Upgrade Survey .....	274

## Chapter 7

<b>Appendices.....</b>	<b>277</b>
Understanding Appendices.....	277

## Appendix A

<b>Applying Fixes Required for Upgrade.....</b>	<b>279</b>
Preparing to Apply Fixes.....	279
Applying Fixes During Installation.....	280
Applying Fixes After Copying Project.....	280
Applying Fixes After Data Conversion.....	281
Applying Fixes Between Upgrade Passes.....	281
Applying Fixes in Move to Production.....	282

## Appendix B

<b>Changing the User Interface.....</b>	<b>283</b>
Change the User Interface.....	283
Change the User Interface for Mobile.....	285

## Appendix C

<b>Improving Performance.....</b>	<b>287</b>
Understanding Performance Improvement.....	287
Planning Your Upgrade for Performance.....	287
Performing the Upgrade.....	287
Verifying the Database Structure Task.....	288
Creating and Altering SQL Scripts.....	288
Performing Data Conversion.....	288
Improving Performance for Your Platform.....	288
Using Oracle.....	288
Using DB2.....	289
Consulting the Global Support Center.....	289

## Appendix D

<b>Modifying the Approval Rule Set Program.....</b>	<b>291</b>
Modify Approval Rule Set Related Tables.....	291

## Appendix E

<b>Planning for Upgrade Testing.....</b>	<b>295</b>
Understanding Testing Techniques.....	295
Deciding When to Test.....	295
Evaluating Your Testing Requirements.....	296

Defining Your Testing Strategy.....	296
Understanding Your Testing Strategy.....	296
Evaluating Unit Testing .....	297
Evaluating System Testing.....	297
Evaluating Integration Testing.....	297
Evaluating Parallel Testing.....	298
Evaluating Performance Testing.....	298
Evaluating User Acceptance Testing.....	298
Evaluating Regression Testing.....	299
Determining the Testing Conditions.....	299
Developing Your Test Plan.....	299
Developing Test Scripts.....	300
Reviewing Tips and Techniques.....	301
Reducing the Time of Upgrade Process.....	301
Performing Security Testing.....	302
Performing Testing on Up to Date Data.....	302
Performing Test Move to Production.....	302
Tracking Issues.....	302
Reviewing Testing Tools.....	302
Discussing Change Control.....	303
Discussing Back Up Procedures.....	303
Evaluating Unexpected Results.....	303
Evaluating Reasons for Failure.....	303

## **Appendix F**

<b>Preserving Queries and Tree Objects.....</b>	<b>305</b>
Understanding Preserve Queries and Trees.....	305
Preparing the Database.....	306
Creating a New Project.....	306
Comparing the New Project.....	307
Copying the Project.....	308
Testing the Project.....	308
ReExporting the PeopleTools Tables.....	308

## **Appendix G**

<b>Reviewing Batch Program Changes.....</b>	<b>311</b>
Reviewing Batch Program Changes.....	311

## Appendix H

<b>Reviewing Benefits Changes.....</b>	<b>313</b>
Reviewing Consolidation of FSA Balances.....	313
Understanding Changes to Snapshot Tables.....	314
Reviewing Savings Management Under EGTRRA of 2001.....	314
Reviewing COBRA Administration.....	315
Reviewing Benefit System Indicator on Job.....	315
Reviewing Benefit Record Number Display.....	315
Reviewing Effective Dating of Dependents.....	316
Reviewing Effective Dating of Company Cars.....	316
Understanding Benefit Program Enhancement.....	316
Reviewing Changes to Relationship Usage.....	317
Understanding Changes to Relationship Usage.....	317
Reviewing Cobra Event Rules.....	317
Reviewing Dependent Relationship Rules.....	317
Reviewing HIPAA Changes.....	317
Reviewing Spending Account Changes.....	317
Understanding Spending Account Setup.....	318
Reviewing Changes to Spending Account Pledge Limits.....	318
Reviewing FSA Claims Processing Setup .....	318
Reviewing Consolidation of Benefit Rates.....	319
Reviewing Consolidation of Coverage Calculations.....	320

## Appendix I

<b>Reviewing HRMS Changes.....</b>	<b>323</b>
Reviewing Record and Field Conversions.....	323
Understanding Record and Field Conversions.....	323
Status Conversions.....	323
Reviewing Profile Management Table Deletions.....	328
Reviewing Records with Duplicate Key Structures.....	328
Understanding Applicant Data Normalization.....	330
Reviewing the National ID Expiration Record.....	331
Reviewing Schedule Defaults.....	332
Reviewing Shift and Workdays Upgrade.....	332
Determining the Effective Date of Schedule Definition.....	332
Reviewing Schedule Override Tables.....	333

**Appendix J**

<b>Sizing Tables for the Upgrade.....</b>	<b>335</b>
Sizing Tables.....	335

**Appendix K**

<b>Understanding Core Record Modifications.....</b>	<b>345</b>
Understanding Core Record Modifications.....	345

**Appendix L**

<b>Understanding Dynamic ChartFields Changes.....</b>	<b>351</b>
Understanding Dynamic ChartFields Upgrade.....	351
Understanding Changes in the Account Code Table.....	351
Understanding Combination Codes.....	351
Understanding FDM_HASH.....	352
Understanding Changes in GL Interface Setup.....	353
Understanding GL Interface Setup.....	354
Understanding Liability Accounts.....	354
Understanding GL Activity Grouping.....	357
Understanding Expenses ChartField Mapping.....	362
Upgrading Time and Labor ChartFields.....	367

**Appendix M**

<b>Understanding FTE Enhancement.....</b>	<b>369</b>
Understanding FTE Enhancement.....	369

**Appendix N**

<b>Understanding Garnishments Changes.....</b>	<b>371</b>
Understanding Garnishments Upgrade.....	371
Understanding Payee Changes.....	371
Understanding Changes to Payee Data.....	372
Updating Payee Data During the Initial Pass.....	372
Updating Payee Data During Move to Production.....	373
Understanding Rules Changes.....	373
Understanding Changes to Garnishment Rules.....	374
Updating Rules During the Initial Pass.....	374
Updating Rules During Move to Production.....	375

**Appendix O**

<b>Understanding Performance Management.....</b>	<b>377</b>
Understanding Performance Management Changes.....	377
Understanding Employee Review Tables .....	377
Understanding Employee Review Tables Changes.....	377
Understanding Employee Review Records.....	378
Understanding Performance Improvement Plan (USF) Records.....	378
Understanding Merit Scale Setup Records.....	378
Understanding Rating Scale Setup Records.....	378
Understanding Review Band Setup Records.....	378
Understanding Performance Plan (USF) Setup Records.....	378
Understanding Employee Review Data .....	379

**Appendix P**

<b>Understanding Person Model Changes.....</b>	<b>381</b>
Understanding Changes.....	381
Understanding Entity Relationships.....	382
Understanding PeopleSoft 8.0 Entity Relationships.....	382
Understanding PeopleSoft 8.8 Entity Relationships.....	383
Understanding PeopleSoft New Release Entity Relationships.....	384
Understanding Record and Table Changes.....	388
Understanding Record Changes.....	388
Understanding Table Changes.....	404

**Appendix Q**

<b>Understanding Service Indicator Changes.....</b>	<b>409</b>
Reviewing Service Indicator Code Upgrade Defaults.....	409
Understanding Basis of Impact.....	410
Analyzing Current Service Impacts and Associations.....	411
Understanding Service Indicator Audits.....	413
Understanding Service Indicator Display Security.....	413

**Appendix R**

<b>Upgrading the Content Provider Registry.....</b>	<b>415</b>
Understanding the Content Provider Registry Upgrade.....	415
Copying Your Enterprise Portal Database.....	417
Upgrading Enterprise Portal PeopleTools.....	417



Updating Registry Permission Lists.....	417
Understanding Registry Permission Lists Updates.....	417
Updating Portal Registry.....	418
Deleting the Database Cache.....	418
Creating the Portal Project.....	418
Understanding Portal Project Creation.....	419
Creating the Target Enterprise Portal Project.....	419
Cleaning the Target Enterprise Portal Project.....	420
Deleting the Target Enterprise Portal Database Cache.....	420
Copying the Target Enterprise Portal Project Definition.....	421
Creating the Copy of Production Portal Project.....	421
Cleaning the Copy of Production Portal Project.....	424
Deleting the Copy of Production Database Cache.....	424
Comparing the Portal Project.....	424
Reviewing the Portal Project.....	425
Copying the Portal Project.....	425
Understanding Portal Project Copying.....	425
Copying the Portal Project.....	426
Deleting the Enterprise Portal Database Cache.....	426
Copying Portal Project to Production.....	426
Understanding Portal Project to Production Copying.....	427
Copying the Portal Project to File.....	427
Copying the Portal Project from File.....	427
Deleting the Enterprise Portal Database Cache Again.....	428
Deleting Obsolete Folders.....	428
Understanding Obsolete Folder Deletion.....	428
Deleting Obsolete Folders on Enterprise Portal 8.4.....	428
Deleting Obsolete Folders on Enterprise Portal 8.8.....	429
Updating Registry Folder Permissions.....	429
Understanding Registry Folder Permissions Updates.....	429
Updating the Enterprise Portal Registry Folder Permissions.....	429
Deleting the Enterprise Portal Cache.....	430

## Appendix S

<b>Upgrading with Tax Updates.....</b>	<b>431</b>
Upgrading with Tax Updates.....	431

## Appendix T

<b>Using Data Conversion Utilities.....</b>	<b>433</b>
Understanding Data Conversion Utilities.....	433
Using the Upgrade Driver Program.....	433
Using the Upgrade Drivers Page.....	433
Understanding Upgrade Drivers Page.....	433
Accessing the Upgrade Drivers Page.....	434
Adding New Upgrade Drivers Section Page.....	434
Inactivating Upgrade Drivers Section.....	435
Reviewing the Data Conversion Report.....	436

## Appendix U

<b>Using the Comparison Process.....</b>	<b>437</b>
Understanding the Comparison Process.....	437
Reviewing Source and Target Columns.....	438
Reviewing the Action Column.....	439
Reviewing the Upgrade Column.....	439
Putting it All Together.....	440
Understanding Upgrade Compare Reports.....	440
Reviewing Report Columns.....	441
Using Reports.....	441

## Appendix V

<b>Using the Data Archiving Tool.....</b>	<b>445</b>
Data Archiving Tool.....	445

<b>Index .....</b>	<b>447</b>
--------------------	------------

# About This Documentation

This preface discusses:

- Understanding This Documentation
- Prerequisites
- Audience
- Organization
- Related Information

---

## Understanding This Documentation

This documentation is designed to direct you through the process of upgrading to your new PeopleSoft release.

This section describes information you should know before you begin working with Oracle's PeopleSoft Enterprise products and documentation, including specific documentation conventions.

---

## Prerequisites

You must complete the tasks in the document *Getting Started on Your PeopleSoft Upgrade* before beginning this upgrade. If you have not yet completed these tasks, do so now.

See Oracle/PeopleSoft Customer Connection (Implement Optimize + Upgrade, Upgrade Guide, Upgrade Documentation and Software, Upgrade Documentation and Scripts).

---

## Audience

This documentation assumes you have a basic understanding of the PeopleSoft system. One of the most important components of a successful upgrade of your PeopleSoft installation is your onsite experience.

You should be familiar with your operating hardware environment and have the necessary skills to support that environment. You should also have a working knowledge of:

- SQL and SQL command syntax.
- PeopleSoft system navigation.
- PeopleSoft windows, menus, and pages (and how to modify them).
- The World Wide Web.
- Microsoft Windows or Windows NT graphical user interface.

PeopleSoft recommends that you complete training before performing this upgrade.

See Oracle/PeopleSoft Customer Connection (Oracle University).

---

## Organization

This documentation is divided into chapters, each containing tasks that represent major milestones in the upgrade process:

- Prepare Your Database
- Apply PeopleTools Changes
- Run and Review Compare Reports
- Apply Application Changes
- Complete Database Changes
- Apply Changes to Production Database

This documentation may also contain appendices. When additional information is required to complete an upgrade task, you will be directed to the appropriate appendix.

---

## Related Information

Oracle provides additional information that may help with your upgrade. The following information (as well as updates to our upgrade) is available on Oracle/PeopleSoft Customer Connection:

- *PeopleSoft Release Notes.* Before you begin your upgrade, read the PeopleTools and PeopleSoft application release notes to determine what has changed in the system and to familiarize yourself with the new features in this release. The release notes also indicate whether you need to upgrade other portions of your system, such as your RDBMS software or batch files.
- *Upgrades Documentation.* The upgrade documentation on Customer Connection contains information posted after shipment of this release that may not be included in these upgrade instructions. This information may include updates and fixes required at upgrade. Always check the upgrade documentation for the most current documentation and information.
- *Getting Started on Your PeopleSoft Upgrade.* Before beginning a PeopleSoft upgrade, you must complete the tasks in the document *Getting Started on Your PeopleSoft Upgrade*. This document guides you through planning your upgrade as well as installing the software necessary to upgrade to the next PeopleSoft release. If you did not complete the tasks in this documentation, do so now.

---

**Important!** Before upgrading, it is imperative that you check the upgrade documentation on Customer Connection for updates to the upgrade instructions. We continually post updates as we refine the upgrade process.

---

See Oracle/PeopleSoft Customer Connection (Implement Optimize + Upgrade, Upgrade Guide, Upgrade Documentation and Software, Upgrade Documentation and Scripts, Release, Enterprise).

# CHAPTER 1

## Prepare Your Database

This chapter discusses:

- Understanding Database Preparation
- Understanding Your Upgrade
- Copying Your Production Database
- Applying Upgrade Planning Files
- Editing Upgrade Planning DB2 Scripts
- Running Upgrade Planning Scripts
- Updating Statistics
- Running Initial Audit Reports
- Running Application Audits
- Preparing Approvals for Upgrade
- Preparing the Benefits Upgrade
- Making Functional Decisions
- Reporting Row Count for Tables
- Preparing Your Database
- Backing Up Demo Databases
- Renaming Records and Fields
- Comparing Customizations
- Identifying Customizations
- Preparing for the Application Upgrade
- Backing Up After Prepare Your Database

---

## Understanding Database Preparation

You must make a copy of your production database before you start preparations for the technical portion of the upgrade. Unless otherwise noted, run these tasks on your Copy of Production database (not the New Release Demo database). These tasks do not use the new PeopleSoft release. You should use your current codeline and current Oracle's PeopleSoft PeopleTools release to perform these tasks unless instructed otherwise.

---

**Important!** You must read the documentation *Getting Started on Your PeopleSoft Upgrade* before you continue with your upgrade. This getting started guide explains the upgrade process, terminology, and setup tasks that *must* be performed prior to starting your upgrade.

---

---

## Task 1-1: Understanding Your Upgrade

This section discusses:

- Understanding PeopleSoft Upgrades
- Verifying the Software Installation
- Defining Upgrade Databases
- Reviewing Upgrade Notes and Tips

### Understanding PeopleSoft Upgrades

This task reviews information you need to know before you begin your upgrade. It explains the different types of databases that you will use and provides useful upgrade tips and information that you may need to apply before beginning your upgrade.

#### Task 1-1-1: Verifying the Software Installation

Before continuing with the upgrade, you must complete all of the tasks in *Getting Started on Your PeopleSoft Upgrade*, “Install the Software Release.” Verify that the following tasks are complete:

- Installing the new release.
- Applying PeopleTools patches.
- Applying updates required for upgrade.
- Installing the Change Assistant.
- Creating and configuring an upgrade job.
- Setting Configuration Manager profile.
- Reviewing upgrade step properties.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

#### Task 1-1-2: Defining Upgrade Databases

The following databases will be used during your upgrade:

- The New Release Demo database always refers to the database delivered with your new PeopleSoft release.

It contains the new and changed database objects that you want to add. The New Release Demo database is also referred to as the Demo database later in the upgrade.

- The Copy of Production database refers to the copy of your production database, into which you will add the new and changed objects for this release from the New Release Demo database.

---

**Note.** You will create more than one Copy of Production database. Your second and subsequent copies are referred to as the New Copy of Production.

---

- The Copy of Current Demo refers to the copy of the demo database for the release that you are currently using.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-1-3: Reviewing Upgrade Notes and Tips

This section contains information that may apply to your upgrade product. Review the information in this section before beginning your upgrade.

- Performance Recommendations

Before beginning your upgrade, you should plan for performance issues.

See Appendix: “Improving Performance.”

- Tax Update Status for Payroll Sites

PeopleSoft HRMS 9.0 is current through Tax Update 06-D.

Oracle recommends that you run your upgrade using your Copy of Production at the same Tax Update level as the HRMS New Release Demo database. If your New Release Demo database is not at the same Tax Update level as listed above, follow the recommendations in the appendix “Upgrading with Tax Updates.”

---

**Note.** If you are unsure of what tax update level you are on, from your Copy of Production, select Define Business Rules, Define Payroll Taxes, Report, Tax Update, and note the latest that is listed.

---

- Making Changes to Company, Company State Taxes, Company Local Taxes, Wage Loss Plan Table, and Deduction Table During the Upgrade

If you are using non-commitment accounting in Payroll for North America, after you create your Copy of Production database in the initial pass, do not make any changes in your production database for Company, Company State Taxes, Company Local Taxes, Wage Loss Plan Table, and Deduction Table until you complete the upgrade. If you make any changes to these tables, you will not be able to use the import script during Move to Production pass to import data that you set up in the initial pass. You will either need to synchronize the data between production and copy of the production in the initial pass before you set up data and do the export, or manually set up data in your copy of the production during the Move to Production pass.

If you are using Time and Labor and integrate with PeopleSoft projects, then you must read and follow the impact of the ChartField upgrade in the appendix “Understanding Dynamic ChartFields Changes.”

- Upgrading Approvals for Job Openings and Job Offers

This upgrade will not upgrade the approvals for job openings and job offers. To upgrade your approvals after completing this upgrade, see Customer Connection for the Approval Documentation. If you do not want to use the documentation from Customer Connection, all job openings and job offers must already be approved or not started before beginning this upgrade.

- Reviewing HRMS Record Changes

Many record changes in the new PeopleSoft release will affect the data in your system. These changes will be made through data conversion. To fully understand these changes, please review the appendixes referenced below.

See Appendix: “Reviewing Benefits Changes.”

See Appendix: “Reviewing HRMS Changes.”

See Appendix: “Understanding Core Record Modifications.”

See Appendix: “Understanding FTE Enhancement.”

See Appendix: “Understanding Performance Management.”

See Appendix: “Understanding Person Model Changes.”

- Upgrading Time and Labor Rules

Your custom Time and Labor Rules will be manually recompiled during your initial upgrade pass only. You should not create any new custom rules in your production database after you have created your Copy of Production database for the initial pass until after you have completed your final Move to Production and have gone live with the new PeopleSoft release. If you do create new Time and Labor Rules in production, you should manually create them again in your upgraded Copy of Production database from the previous pass before starting your next pass. This is the database that will be the source database for your Move to Production pass.

- Microsoft SQL Server Column Statistics

As of Microsoft SQL Server 2000, user-defined statistics can be created on columns within a table. This feature is not supported by PeopleTools. If you added user-defined statistics to any columns in your PeopleSoft application, it may cause errors to occur during the upgrade steps that alter tables. PeopleSoft recommends that you drop all user-defined statistics on columns of PeopleSoft tables before proceeding with your upgrade.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 1-2: Copying Your Production Database

This section discusses:

- Making a Copy of Production Database
- Increasing Database Space



## Task 1-2-1: Making a Copy of Production Database

Make a copy of your production database. You will upgrade this database, rather than performing the upgrade directly on your production database. Performing an upgrade on a copy of your production database enables you to test your upgrade in a controlled environment. Refer to the administration guide for your database platform for information on copying databases.

---

**Note.** Move to Production: This is a second Copy of Production sometimes referred to as the New Copy of Production. The first Copy of Production, or “old” Copy of Production, will now be the source database (it was the target database in the initial test pass). The second, or “new” Copy of Production, is now the target database.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-2-2: Increasing Database Space

To prepare for the upgrade, you may need to increase the space allocated to your Copy of Production database. Depending on your relational database management system (RDBMS), this may include allocating space to tablespaces or allocating database primary space and log files. Be aware that your new environment needs to accommodate both the existing data in your Copy of Production database as well as the new data, new data structures, and new database objects. Every site and configuration is different, so Oracle cannot offer a guaranteed estimate of your database sizing needs.

As part of the initial upgrade pass, you may need to revisit your initial space allocation settings more than once as you progress through the upgrade. At the end of the initial pass, the final space allocation settings will closely reflect the space you will need to complete any subsequent Move to Production passes. Work with your DBA to ensure that your environment is set up appropriately for both the initial and Move to Production passes.

See Appendix: “Sizing Tables for the Upgrade.”

See the PeopleSoft installation documentation for your product line and release.

---

**Note.** Oracle RDBMS customers also need to alter the tablespace for PSIMAGE and increase it to 200 MB; autoextend on next 10 MB; maxsize unlimited.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 1-3: Applying Upgrade Planning Files

This section discusses:

- Understanding Applying Upgrade Planning Files

- Applying the UPGOPT Project
- Building the UPGOPT Project
- Setting Up Upgrade Planning Security

## Understanding Applying Upgrade Planning Files

In this task, you apply the upgrade planning files that you downloaded from the Upgrade database on Customer Connection to your current codeline. These files may include Structured Query Report (SQR) programs and scripts that you will execute in later tasks, and a project that you will apply to your Copy of Production database. This project may include records, fields, pages, menus, queries, and process definitions that allow functional users to define conversion information needed for later upgrade tasks.

### Task 1-3-1: Applying the UPGOPT Project

In this step, apply the UPGOPT project to your Copy of Production database using the Copy Project from File process.

To apply the UPGOPT project:

1. Using your current codeline, launch Application Designer and sign in to your Copy of Production database.
2. Select File, Copy Project from File.
3. From the dialog box, select the import directory PS\_HOME\PROJECTS\ (current codeline).
4. Select UPGOPT in the Project Name box, and then click the Copy button.
5. When the Copy dialog appears, click Options.

On the Copy Options tab, if you see a Filter By Release check box, make sure it is cleared.

Whether you see this will depend on your version of PeopleTools.

6. Click OK to return to the Copy dialog.
7. Click Copy.

This copies the UPGOPT project onto your Copy of Production database.

8. Using your current codeline, launch Data Mover and sign on to your Copy of Production database.
9. Run the following script to load Access Groups and the Upgrade Query Tree:

```
\PS_HOME\SCRIPTS\UPGOPT_HC80.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 1-3-2: Building the UPGOPT Project

In this step you create and alter tables, and create views.

To build the UPGOPT project:

1. Using your current codeline, launch Application Designer and sign in to your Copy of Production database.

2. Select File, Open...
  3. In the Definition drop-down list box, select *Project* and click Open to display the list of projects.
  4. Select *UPGOPT* and click Open again.
  5. Select Build, Project...
  6. Under Build Options, select Create Tables, Create Views.
  7. Click Settings...
  8. On the Create tab, select Recreate View if it already exists and Recreate Table if it already exists.
  9. On the Logging tab, select Fatal errors, warnings, and informational messages.
  10. On the Scripts tab, select Output to separate files.
  11. In the Script File Names box, give your scripts a unique name that reflects this task number and the object being created.
  12. Click OK.
  13. Under Build Execute Options, select Build script file.
  14. Click Build.
  15. Using the appropriate SQL query tool for your platform, run the scripts created in the step above.
- Run the scripts in the following order: Create Tables, Create Views, Create Indexes.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-3-3: Setting Up Upgrade Planning Security

In this step you set up security on your Copy of Production database.

To set up security:

1. Select PeopleTools, Security, Use, Permission Lists.
2. Enter the permission list for the users who will be reviewing and setting up functional requirements for the upgrade, then click Search.
3. On the Pages tab, select or insert the menu *UPG\_DEFINE\_DEFAULTS* and click Edit Components.
4. Select all of the items in the menu.
5. Click OK.
6. Click Save.
7. Select PeopleTools, Security, Permissions and Roles, Permission Lists.
8. Select the permission list for the users that review and set up functional requirements for the upgrade.
9. Select the Query tab.
10. Select Access Group Permissions.
11. Add one row with the tree name *UPG\_QUERY\_TREE* and the access group *UPG\_ACCESS\_GROUP*.

12. Select OK, Save.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-4: Editing Upgrade Planning DB2 Scripts

Perform this step only if your database platform is DB2 z/OS. DB2 z/OS scripts that create tables need a `set current sqlid` statement so that the tables are created with the correct owner ID. Open each script listed below, then uncomment and modify all of the DB2-specific statements to reflect your environment.

For SQL scripts, if the script does not contain DB2-specific statements, add the following line to the top of the script and edit it for your environment, inserting the appropriate owner ID in uppercase characters:

```
set current sqlid = 'OWNER_ID';
```

For Data Mover scripts (DMS), if the script does not contain DB2-specific statements, add the following line to the top of the script and edit it for your environment, inserting the appropriate owner ID in uppercase characters:

```
set execute_sql set current sqlid = 'OWNER_ID';
```

The following is a list of scripts that you need to edit:

```
PUUPX07.DMS
PUHCTLW01.DMS
PUHCHRW01I.DMS
PUHCEOG04DB2.SQL
PUHCHRG10.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 1-5: Running Upgrade Planning Scripts

This section discusses:

- Understanding Running Upgrade Planning Scripts
- Loading Planning Messages and Strings
- Populating Recruiting Solutions
- Populating the Equation Renames

- Importing Upgrade Default Options

## Understanding Running Upgrade Planning Scripts

In this task, you run scripts that will load data needed to complete the task Making Functional Decisions later in the upgrade.

### Task 1-5-1: Loading Planning Messages and Strings

This script loads messages and strings needed for the Upgrade Planning Pages. The script name is PUHCUPG09.DMS and the data file is PUHCUPG09.DAT.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 1-5-2: Populating Recruiting Solutions

PUHCERG10.DMS populates the PS\_UPG\_ER\_STATUS and PS\_UPG\_ER\_STS\_RSN tables with default mappings of the status translate values to translate codes.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 1-5-3: Populating the Equation Renames

This script loads PS\_UPG\_SFAEQCNV, PS\_UPG\_SSFEQCNV, PS\_UPG\_SFAEQCV2 and PS\_UPG\_SSFEQCV2 tables with the default renames for the PeopleSoft 9.0 Demo equations.

The script and data file name for your upgrade is:

PUCSFAG01

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 1-5-4: Importing Upgrade Default Options

In this step, you import the upgrade default options defined during the previous pass for use during Move to Production.

The script name for your upgrade is:

PUHCHRW01I.DMS

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

---

**Task 1-6: Updating Statistics**

Run this task to improve performance of your compare and copy processes. Have your database administrator update statistics on your database before proceeding with your upgrade. Later in the upgrade, you will update your statistics again due to changes in the database structure.

See Appendix: “Improving Performance.”

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

**Task 1-7: Running Initial Audit Reports**

This section discusses:

- Understanding Running Initial Audit Reports
- Run the Initial DDDAUDIT Report
- Run the Initial SYSAUDIT Report
- Run the Initial Alter Audit
- Review the Initial Audits

**Understanding Running Initial Audit Reports**

In this task, you run and review your initial DDDAUDIT, SYSAUDIT, and Alter Audit reports. Running these reports ensures that your database is as clean as possible for the remainder of the upgrade.

**Task 1-7-1: Run the Initial DDDAUDIT Report**

DDDAUDIT is an SQR script that compares your production SQL data tables with the PeopleTools record definitions to identify inconsistencies.

In this step, DDDAUDIT is run using SQR from your current (old) release of PeopleSoft against the Copy of Production to ensure that you are starting with a clean database.

You will review the output from the report in a later step.

See Review the Initial Audits.

See the Enterprise PeopleTools PeopleBook: System and Server Administration for your current release.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-7-2: Run the Initial SYSAUDIT Report

SYSAUDIT is an SQR script used to identify “orphaned” PeopleSoft objects. For example, SYSAUDIT can identify a module of PeopleCode that exists but does not relate to any other objects in the system. SYSAUDIT also identifies other inconsistencies within your database.

In this step, SYSAUDIT is run using SQR from your current (old) release of PeopleSoft against the Copy of Production to ensure that you are starting with a clean database.

You will review the output from the report in a later step.

See Review the Initial Audits.

See the Enterprise PeopleTools PeopleBook: System and Server Administration for your current release.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-7-3: Run the Initial Alter Audit

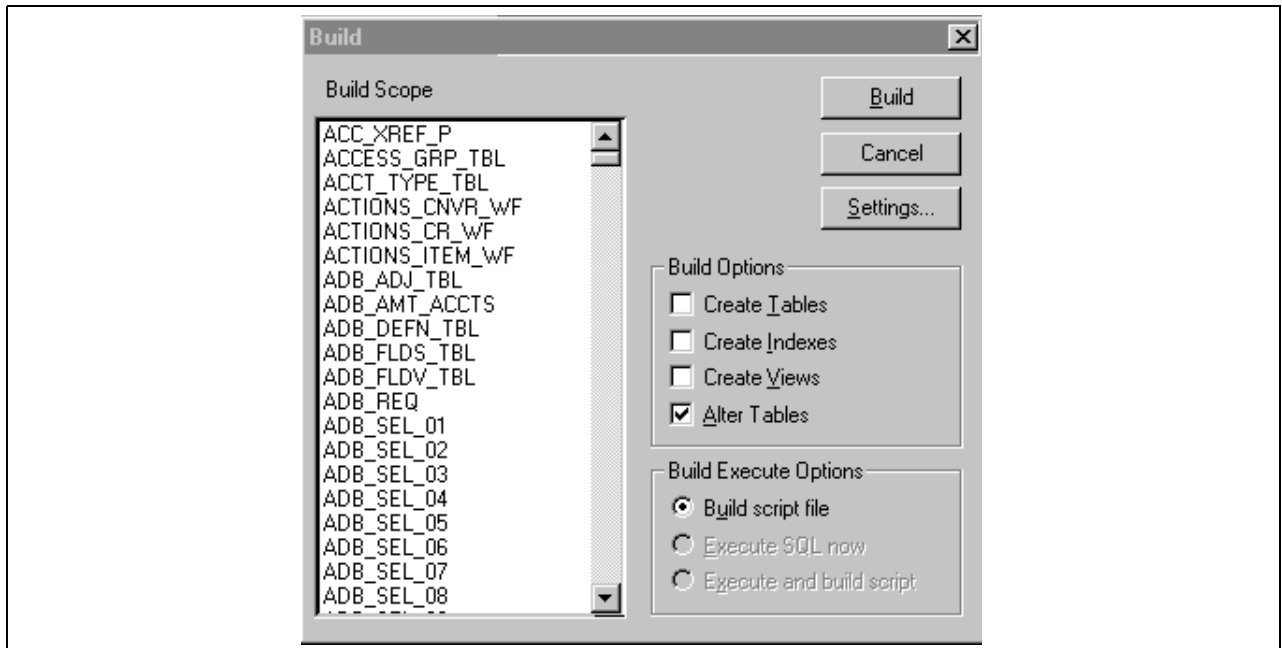
To verify that the PeopleTools definitions are synchronized with the underlying SQL data tables in your database, run the PeopleTools alter record process on all records in your system. This process, called an Alter Audit, compares the data structures of your database tables with the PeopleTools definitions to identify inconsistencies. Alter Audit then creates SQL scripts with the data definition language (DDL) changes that are required to synchronize your database with the PeopleTools definitions.

To run Alter Audit:

1. Launch PeopleTools and sign in to the Copy of Production database.
2. From Application Designer, select File, New...
3. Select Project, and then click OK.
4. Select Insert, Objects into Project...
5. Select Records from the Object Type drop-down list box.
6. Select Table from the Type drop-down list box.
7. Click Insert, and then click Select All.

8. Click Insert, and then click Close.
9. Select File, Save All.
10. Enter a project name of your choice.
11. Click OK.
12. Select Build, Project...

The Build dialog box appears:



Build dialog box

13. Under Build Options, select Alter Tables.  
Create Indexes is automatically selected.
14. Under Build Execute Options, select Build script file.
15. Click Settings...

The Build Settings dialog box appears.

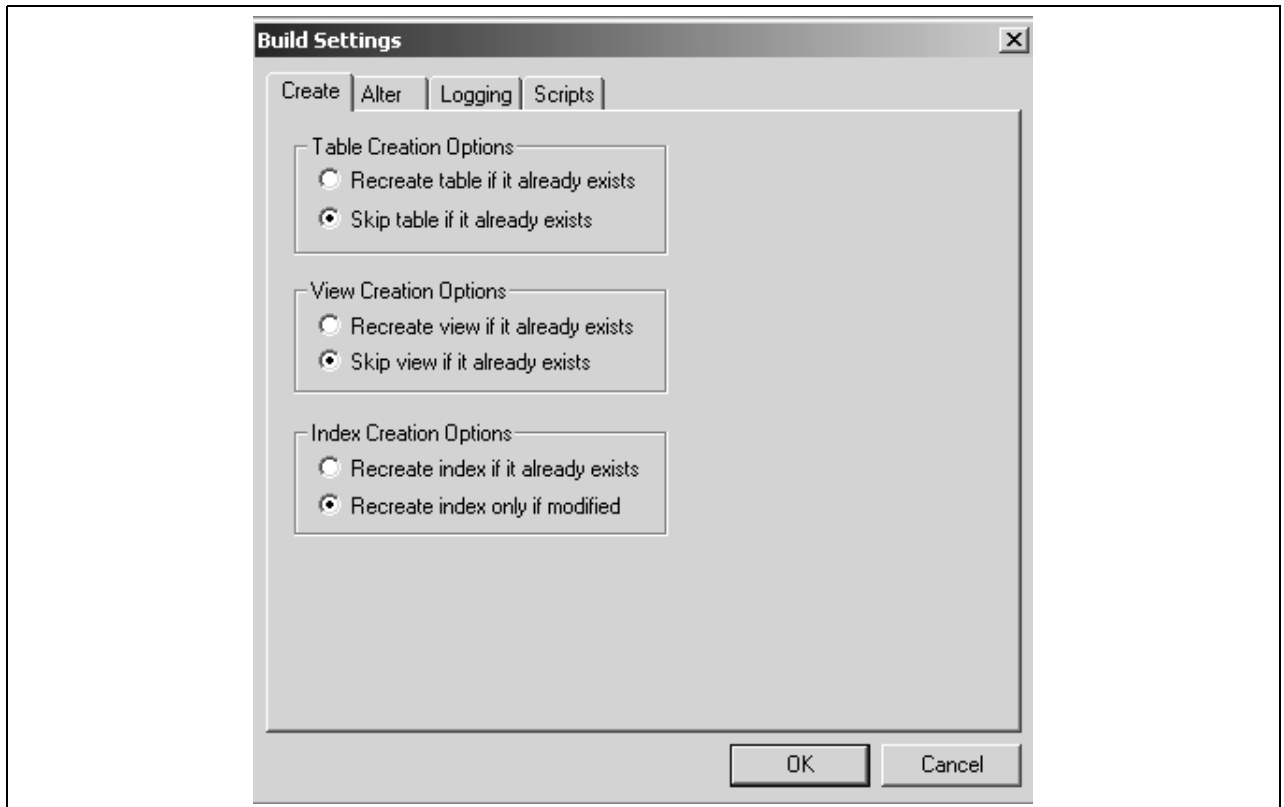
16. Select the Scripts tab.





Build Settings dialog box: Scripts tab

17. Under Comments, select Write Alter comments to script.
18. Under Script File Options, select Output to separate files and enter script names.
19. Select the Create tab.



Build Settings dialog box: Create tab

20. Under Index Creation Options, select Recreate index only if modified.
21. Select the Alter tab.



Build Settings dialog box: Alter tab

22. Ensure that the following fields are set correctly:

- Under the Alter Any option section, the Adds, Changes, Renames, and Deletes check boxes should be selected.
- Drop Column Options should be set to Drop Column if data present.
- Change Column Length Options should be set to Truncate data if field too short.

23. Click OK.

The Build dialog box reappears.

24. Click Build on the Build Dialog.

25. Click Close when complete.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-7-4: Review the Initial Audits

In this step, you review the audits you that performed earlier in this task. Review the audits before proceeding with the upgrade.

Review the output from the SYSAUDIT and DDDAUDIT reports and correct any discrepancies. When application tables are deleted from the Application Designer, they are not automatically deleted from the system tables. Oracle takes this precaution in case you have customized information that you want to preserve. When you review your DDDAUDIT listing, these tables are listed as a discrepancy between the PeopleSoft application and the database.

Now you must decide whether to drop these tables or retain them. In most cases, you will want to drop the tables, using your SQL tool to drop the tables from the system catalogs. If you have customized information or processes that access these tables, you may want to retain them in the system tables even though they will no longer be accessed or updated by the PeopleSoft system. Drop any unnecessary deleted tables now so that your future DDDAUDIT reports will be as clean as possible.

The Alter Audit produces your named scripts from the previous step. These scripts contain SQL that corrects any discrepancies between your PeopleTools Record definitions and the database system catalog table definitions. Review the Alter Audit output and correct any discrepancies.

---

**Note.** You will rerun the DDDAUDIT and SYSAUDIT SQR scripts later in the upgrade. If you want to preserve the log files generated by Change Assistant from this run, you will need to rename the files manually after completing this task.

---

See the Enterprise PeopleTools PeopleBook: System and Server Administration for your current release.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 1-8: Running Application Audits

This section discusses:

- Running Upgrade Reports
- Correcting Garnishment Payee Data
- Preparing for Personal Data Changes
- Running Dynamic ChartFields Audits

### Task 1-8-1: Running Upgrade Reports

This section discusses:

- Running Upgrade Reports
- Running the Audit for Merit Scale Table Report
- Running the Desired Distribution Table Report
- Running the Account Code Table Audit Report
- Running the US Custom Garnishment Rules Audit Report

- Running the Affirmative Action Plan Report
- Running the Person Relationship Audit Report
- Running Jobcode Table Reports

## Running Upgrade Reports

In this step, you run upgrade reports that list erroneous data that you need to clean up before starting the upgrade.

To run the application audit reports:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Report.
3. Select the appropriate reports.

---

**Note.** Reports are identified and explained in the remainder of this task. You will need to perform this step for all reports identified below.

---

4. Select Add.
5. Enter the run control *UPG\_AUDIT*.
6. Click the Run button.
7. Click the Process Monitor link to monitor the program's process.
8. Repeat the steps above for all reports identified in the remainder of this task.

## Running the Audit for Merit Scale Table Report

Run the Audit for Merit Scale Table Report, PUHRG65.SQR.

This report lists Merit Scale Table collisions with Rating Models. For the new PeopleSoft release, the Review Band series defined in the Merit Scale Table (for use in Salary Planning) is being consolidated into the Rating Model structure. To do this, PeopleSoft created new Rating Models for each existing Merit Scale that is based on a Review Band. However, it is possible that a Rating Model already exists with the same name (Key) as the Merit Scale being converted, in which case that Merit Scale (Review Band series) will not be able to upgrade. This report identifies any such points of conflict. If this report identifies any "empty" Rating Model, where a Rating Model description has been defined but no review ratings have been entered, you must delete the empty Rating Model to avoid errors during the upgrade. If the report identifies key-name conflicts you can manually redefine a comparable Rating Model after upgrade is complete, but this is optional.

To delete the empty rating models:

1. From your web browser, sign in to the Copy of Production database.
2. Select Administer Workforce, Plan Salaries (GBL), Setup, Rating Model Table.
3. Enter the Rating Model from the report to be deleted and click OK.
4. Select Edit, Delete Row.

## Running the Desired Distribution Table Report

Run the Dump Desired Distrib Tbl Report, PUHRG62.SQR.

In this PeopleSoft release, the Review Distribution tables (REVIEW\_DISTRIB, REVW\_DESIR\_DIST) used to analyze performance results are being replaced by new structures (EP\_DESIRE\_DIST and EP\_DES\_DIST\_PCT), which integrate better with ePerformance. The existing data in these tables cannot be upgraded due to the wide differences in usage and business flow. This report displays your existing distribution setup in a format to assist you in reestablishing your distribution requirements under the ePerformance structures after the upgrade is complete, if you would like.

## Running the Account Code Table Audit Report

Run the Account Code Table Audit Report, PUPYW01.SQR.

This report lists all account codes that have the following ChartField setup:

- No ChartFields defined.
- Some ChartFields are defined but account has not been defined.

Records without defined ChartFields are not converted during the upgrade. If you have any records defined without defined ChartFields and you want to convert them to the new PeopleSoft release, update ACCT\_CD\_TBL with at least the Account value.

Records with some ChartFields defined but without Account ChartField value will still be converted during the upgrade unless it has not been used for distribution anywhere in PeopleSoft HRMS. However, if you have any records defined like this, you must update ACCT\_CD\_TBL with Account value. Account is a mandatory field in the new PeopleSoft release and conversion without that value cannot complete correctly.

To update the Account Code Table with ChartField values:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Payroll Process, Setup 1, Account Code Table or Define Business Rules, Define Commit Accounting <country>, Setup, Account Code Table, where <country> is, for example, CAN, GBL, or US.
3. Enter the account code from the report.
4. Click Search.
5. Insert new effective date and update ChartFields Details.
6. Click Save.

## Running the US Custom Garnishment Rules Audit Report

Run the U.S. Custom Garnishment Rules Audit Report, PUPYT04.SQR.

This report has the same layout as the Garnishment Rules Report, except it lists only U. S. rules maintained by you.

If there are any rules listed in this report, you have two options:

- You may decide that you no longer need the custom rule, in which case you will have to update all active garnishment orders to replace your rule with another rule later in the upgrade.
- You may decide that you still need your custom rule, in which case you will have to re-create that rule later in the upgrade.

If there are no rules listed in this report, you do not have any U. S. custom garnishment rules and do not have to take any action.

See Appendix: “Understanding Garnishments Changes.”

## Running the Affirmative Action Plan Report

Run the Affirmative Action Plan Rpt, PUHRG67.SQR.

This reports on the data in the AFF\_ACTN\_GOALS, AFF\_ACTN\_PLAN, AFF\_ACTN\_PL\_TBL and AFF\_ACTN\_PL\_LNG tables that will be deleted during the upgrade. This report is for informational purposes only.

## Running the Person Relationship Audit Report

Run the Relationship Validation Report, PUHCG01.SQR.

This report shows invalid relationships among tables PERS\_DATA\_EFFDT, PERSONAL\_DATA, PERSON, JOB, EMPLOYMENT, PA\_RT\_EMP\_SETUP, ASGN\_HOME\_HOST, ASSIGNMENT.

This audit reveals any missing or inaccurate relationships that exist in your current Person and Job data. The success of the conversion process depends on this data being correct.

The actions you must take will depend on the results of your investigation. If data exists in one record without being in another, then one of two things is needed. You should either delete the record from the table in which it appears, or add the record to the table from which it is missing.

If you need to add a missing row, you will need to do some functional research to determine what data should be entered.

The information below will help you correct invalid relationships:

- All EMPLIDs in PERSONAL\_DATA that are not in PERS\_DATA\_EFFDT  
A row is required in PERS\_DATA\_EFFDT for every row in PERSONAL\_DATA.  
Any entry in this report means that a row exists in PERSON but does not exist in PERS\_DATA\_EFFDT.
- All EMPLIDs in PERSONAL\_DATA that are not in EMPLOYMENT  
PERSONAL\_DATA is the Parent record to EMPLOYMENT. A row is required in EMPLOYMENT for every row in PERSONAL\_DATA, unless you are using Campus Solutions.
- All EMPLIDs in PERS\_DATA\_EFFDT that are not in PERSONAL\_DATA  
PERSONAL\_DATA is the Parent record to PERS\_DATA\_EFFDT. A row is required in PERS\_DATA\_EFFDT for every row in PERSONAL\_DATA.
- All EMPLIDs in EMPLOYMENT that are not in PERSONAL\_DATA  
PERSONAL\_DATA is the Parent record to EMPLOYMENT. A row is required in EMPLOYMENT for every row in PERSONAL\_DATA.
- All EMPLID/EMPL\_RCDS in EMPLOYMENT that are not in JOB  
EMPLOYMENT is the Parent record to JOB. A row is required in JOB for every distinct EMPLID/EMPL\_RCD in EMPLOYMENT.
- All EMPLID/EMPL\_RCDS in JOB that are not in EMPLOYMENT  
EMPLOYMENT is the Parent record to JOB. A row is required in EMPLOYMENT for every distinct EMPLID/EMPL\_RCD in JOB.
- All EMPLIDs with PER\_TYPE = P in the PERS\_DATA\_EFFDT field that are not in PA\_RT\_EMP\_SETUP as EMPLID or EMPLID\_PAYEE  
PER\_TYPE = P is for Pension non-employees. Anyone with this PER\_TYPE should also be in PA\_RT\_EMP\_SETUP. If they are not, then the PER\_TYPE is probably incorrect and should be fixed before running the conversion.

- All EMPLID\_PAYEE /EMPL\_RCD\_PAYEE in PA\_RT\_EMP\_SETUP that are not in EMPLOYMENT  
Any record in PA\_RT\_EMP\_SETUP should have a corresponding row in EMPLOYMENT. This is when the EMPLID is in the EMPLID\_PAYEE field in PA\_RT\_EMP\_SETUP. If not, then they will not be correctly processed as Pension Payees. Determine whether the Pension data is correct and create Employment/JOB/JOB\_JR data if it is. Otherwise, delete the Pension data.
- All EMPLID/EMPL\_RCD in PA\_RT\_EMP\_SETUP that are not in EMPLOYMENT  
Any record in PA\_RT\_EMP\_SETUP should have a corresponding row in EMPLOYMENT. This is when the EMPLID is in the EMPLID field in PA\_RT\_EMP\_SETUP. If not, then they will not be correctly processed as Pension Payees. Determine whether the Pension data is correct and create Employment/JOB/JOB\_JR data if it is. Otherwise, delete the Pension data.
- All EMPLID/EMPL\_RCDs flagged as HOST records that are not in the Global Assignment table PS\_ASGN\_HOME\_HOST  
This report shows all EMPLID/EMPL\_RCDs flagged as HOST records that are not in the Global Assignment table PS\_ASGN\_HOME\_HOST. Either create the Assignment and Assignment Home/host records or change the value in the HOME\_HOST\_CLASS field to *M* in the EMPLOYMENT table.
- All EMPLID/EMPL\_RCDs flagged as HOST records that are not in the Global Assignment table PS\_ASSIGNMENT  
This report shows all EMPLID/EMPL\_RCDs flagged as HOME records that are not in the Global Assignment table PS\_ASSIGNMENT. Either create the Assignment and Assignment Home/host records or change the value in the HOME\_HOST\_CLASS field to *M* in the EMPLOYMENT table.
- All EMPLID/EMPL\_RCDs flagged as HOST records that were created with HIRE not ASG  
Host Records must have been created with the Action ASG. These records should be changed to have the EMPLOYMENT.HOME\_HOST\_CLASS field set to *M*.

## Running Jobcode Table Reports

Run the Jobcode EEO Group Reports, PUHRG75.SQR.

In previous PeopleSoft releases, the Job Code Table did not inherit values from the EEO Job Group Table for the EEO Job Group field, but the Job Code Table does inherit these values in the new PeopleSoft release. Report PUHRG75.SQR contains two reports that print the EEO Job Groups, which exist in the Job Code Table. The first report indicates the SETID, Jobcode, Effective Date, Jobcode Description, and EEO Job Group for each row within the Job Code Table. The second report indicates the unique values for each EEO Job Group that is found within the Job Code Table. For each unique EEO Job Group that is found during the conversion process, a row will be inserted into the EEO Job Group Table. The value listed for EEO Job Group will be used for the Description and Short Description when the row is inserted.

You can limit the number of different values that will be inserted into the EEO Job Group Table by updating the Job Code Table now. You can then run the report again to determine whether there are any remaining rows that will be inserted during data conversion.

To update the Job Code Table:

1. From your web browser, sign on to your Copy of Production database.
2. Select Define Business Rules, Manage Human Resources(GBL), Setup, Job Code Table.
3. On the Job Code Profile tab, enter the job code you want to update and click Search.
4. Enter *EEO Job Group* in the Job Code field.
5. Click Save.



## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-8-2: Correcting Garnishment Payee Data

This section discusses:

- Understanding Garnishment Payee Data
- Running the Garnishment Payee Audit Report
- Updating Garnishment Orders
- Updating Payee Information in the Payee Table

### Understanding Garnishment Payee Data

If you are using AP Interface and running the Extract Non-Tax Deductions program, you have to ensure that the payee defined in the active garnishment order is set up in the Payee Table. The extract program will not be able to create an invoice if the payee is missing from the Payee Table.

To verify garnishment payee data, you should run the Garnishment Payee Audit report.

The report has two sections:

- Active garnishment orders (orders with the status Approved or Received)
- Inactive garnishment orders (orders with any status other than Approved or Received)

If there are any active orders listed in the report, you will have to correct the data. You can correct the data either by correcting the payee data on the garnishment order, or you can add payee information to the Payee Table.

If there are any inactive orders listed in the report, you can decide either to leave the data as is, or to correct the data. Leaving inactive orders without updating them is acceptable. Inactive orders will not be processed after the upgrade; therefore, you will not need to create an invoice. However, if you decide to correct the history, you can proceed with updates the same way as for active orders.

If there are no orders listed in the report, your data is clean and you do not have to take any action.

See Appendix: “Understanding Garnishments Changes.”

### Running the Garnishment Payee Audit Report

Run the Garnishment Payee Audit Report, PUPYT03.SQR.

This report lists all garnishment orders with a payee that does not exist in the Payee Table.

To run the Garnishment Payee Audit Report:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Report.
3. Select the Garnishment Payee Audit Report
4. Click the Run button.
5. Click the Process Monitor link to monitor the program’s process.

## Updating Garnishment Orders

To update an order with the correct payee data:

1. From your web browser, sign in to the Copy of Production database.
2. Select Compensate Employees, Maintain Payroll Data <country>, Use, Garnishment Spec Data, where <country> is, for example, USF, CAN, or USA.
3. Enter the employee ID and the company from the report.
4. Click Search.
5. If there is more than one garnishment order for the same employee, select the row with the garnishment ID from the report.
6. Click the Garnishment Spec Data 2 tab.
7. Update any of the following fields: SetID, VendorID, or Address.
8. Click Save.
9. Repeat steps 3 – 8 for each garnishment order you want to update.

## Updating Payee Information in the Payee Table

To add payee information to the Payee Table:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Payroll Process, Setup 1, Garnishment Payee Table.
3. Select the Add a New Value tab.
4. Enter the SetID and VendorID from the report.
5. Click Add.
6. Enter all required information for the payee you are setting up.
7. Click Save.
8. Repeat steps 3 – 7 for each payee you want to add.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 1-8-3: Preparing for Personal Data Changes

This section discusses:

- Understanding Personal Data Changes Report
- Setting Up Variable Compensation
- Removing/Replacing Group Build Where Clause
- Updating Time and Labor
- Running Personal Data Reports

## Understanding Personal Data Changes Report

The PUHCG51.SQR for Personal Data Changes runs to identify the areas where you must take action prior to the upgrade. This SQR identifies the data and records where the record PERSONAL\_DATA was used as a field value and cannot be modified by the upgrade process. The following subtopics describe the four sections found in the report.

See Appendix: “Understanding Person Model Changes.”

## Setting Up Variable Compensation

The Variable Compensation section lists the Variable Compensation Formula that has a different PERSONAL\_DATA field than YEARS\_OF\_EXP. All fields from PERSONAL\_DATA need to be removed or replaced. During the upgrade process, any reference to PERSONAL\_DATA.YEARS\_OF\_EXP to PERS\_DATA\_EFFDT.YEARS\_OF\_EXP will be converted. The Variable Compensation section of the report can be viewed in the VC\_PAYFORM\_TBL that uses the Record PERSONAL\_DATA and a field other than YEARS\_OF\_EXP. You will need to correct these manually, as the presence of these fields in the payform table is a result of a customization. Follow the steps below to manually set up the variable.

To manually set up the variable:

1. From your web browser, sign in to your Copy of Production database.
2. Select Compensate Employees, Manage Variable Compensation, Setup, Payout Formulas.

## Removing/Replacing Group Build Where Clause

The Group Build section lists the groups in which a field from the PERSONAL\_DATA record is used in the Group Build Where Clause, and not in the PERSON or PERS\_DATA\_EFFDT. All fields from the PERSONAL\_DATA record need to be removed or replaced. During the upgrade process, any field from the PERSONAL\_DATA record that is in PERSON or PERS\_DATA\_EFFDT will be converted. You will need to delete or modify them online. Follow the steps below to remove or replace the Group Build Where Clause.

To manually remove or replace the Group Build Where Clause:

1. Using your current code line, launch Application Designer and sign in to your Copy of Production database.
2. Select one of the following as appropriate:
  - a. Define Business Rules, Manage Human Resources, Setup, Group Definition.
  - b. Define Business Rules, Manage Human Resources, Setup, Group Criteria Definition.
  - c. Compensate Employees, Manage Variable Compensation, Setup, VC Group Build.
3. Continue to remove or replace the Group Build Where Clause.

## Updating Time and Labor

The Time and Labor (TL) section lists all the PERSONAL\_DATA fields used in TL Group Definition. All the fields need to be replaced or removed. During the upgrade process, PERSONAL\_DATA will be replaced with PERSONAL\_DT\_FST for fields that exist in that view; others will be deleted. You will need to determine whether you want to replace the entries for the Address or Phone fields with different tables prior to performing the upgrade. Follow the steps below to manually update Time and Labor.

To manually update Time and Labor:

1. From your web browser, sign in to your Copy of Production database.
2. Select one of the following:
  - a. Define Business Rules, Define Time and Labor, Enroll, Group Creation Security.

- b. Define Business Rules, Define Time and Labor, Enroll, Dynamic Group.
- c. Define Business Rules, Define Time and Labor, Enroll, Static Group.

## Running Personal Data Reports

Follow the steps below to run personal data reports.

To run the personal data reports:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Reports, Personal Data Reports, Add.
3. Enter the run control *UPG\_PERSONAL\_DATA*.
4. Click the Run button.
5. Click the Process Monitor link to monitor the program's process.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-8-4: Running Dynamic ChartFields Audits

This section discusses:

- Understanding Dynamic ChartFields Audit Reports
- Running the Dynamic ChartFields Audit Reports

### Understanding Dynamic ChartFields Audit Reports

In this task, you run and review the Dynamic ChartFields Audit reports. Running these reports helps you decide how you want to set up data for the General Ledger Interface.

---

**Note.** Run this step only if you are using non-commitment accounting in Payroll for North America.

---

There are four reports, listed below:

- ChartKey Usage by Company and Business Unit: This report indicates the Company Earnings Accounts, Company Deduction Accounts, US Company Tax Accounts, and Canadian Company Tax Accounts.
- ChartKey Usage by Expense Type: This report shows Earnings Expense Accounts, Deduction Expense Accounts, US Tax Expense Accounts, and Canadian Tax Expense Accounts.
- Liability ChartField Usage: This report displays Local Tax Liability ChartFields and Deduction Liability ChartFields.
- List Payroll-GL Chart Key Mappings: This is an existing report and shows all ChartField records.

The reports list the existing setup for expenses and liability accounts. You should work with the General Ledger staff to decide how to group payroll activities, what ChartField combination to use for each group, and what liability accounts to use to offset the credit transactions for payroll activities. After making those decisions, you will manually set up the data in tables used for the General Ledger Interface. You will perform this task later in the upgrade process.

See “Complete Database Changes,” Setting Up General Ledger Interface.

See Appendix: “Understanding Dynamic ChartFields Changes.”

## Running the Dynamic ChartFields Audit Reports

To run the reports:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Report, Chart Field Audit Reports.
3. Select Add a New Value.
4. Enter the run control *UPG\_PUPYG01\_RUN*.
5. Click Add.
6. Click Run.
7. Select all reports.
8. Click OK.
9. Click the Process Monitor link to monitor the program’s process.
10. Select Compensate Employees, Administer GL Interface, Report, Chartkey Values.
11. Select Add a New Value.
12. Enter the run control *Chartkey*.
13. Click Add.
14. Click Run.
15. Click OK.
16. Click the Process Monitor link to monitor the program’s process.

See “Complete Database Changes,” Setting Up General Ledger Interface.

See Appendix: “Understanding Dynamic ChartFields Changes.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

---

## Task 1-9: Preparing Approvals for Upgrade

This section discusses:

- Understanding Approvals Upgrade
- Completing eProfile Approvals

## Understanding Approvals Upgrade

You will have to complete all ePerformance approval transactions and eProfile Promotions, transfers, and reporting changes currently in process before upgrading to the new Approvals Framework.

### Task 1-9-1: Completing eProfile Approvals

You need to complete approvals, promotions, transfers, and reporting changes currently in process before upgrading to the new release. In this step, you run a series of queries to determine which transactions need to be completed and then complete them. If you do not complete these transactions, the data cannot be added to the new approval framework. For instructions on completing these transactions, see your current release eProfile PeopleBook.

To run the eProfile Approvals query reports:

1. Select PeopleTools, Query Manager, Use, Query Manager.
2. Run the following query reports:
  - UPG\_EL\_PROMOTIONS to determine which promotions need to be completed or cancelled.
  - UPG\_EL\_TRANSFERS to determine which transfers need to be completed or cancelled.
  - UPG\_EL\_REPORTINGS to determine which reporting changes need to be completed or cancelled.

See *PeopleSoft 8.00.01 eProfile PeopleBook*.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	eProfile	All	All

---

## Task 1-10: Preparing the Benefits Upgrade

This section discusses:

- Preparing Benefits Administration
- Preparing COBRA Administration

### Task 1-10-1: Preparing Benefits Administration

This step is performed by the Benefits Administrator. Run this step if you are licensed for PeopleSoft Benefits Administration.

You need to create events for all pending data changes that could affect Benefits eligibility because data conversion cannot convert pending data change "triggers." To create events, perform a single Event Maintenance (EM) run (using any single arbitrary existing EM schedule; the smaller the schedule the less impact this will have on your timeline). Then review all Benefits Administration messages to ensure that all pending activity has been successfully processed to events in at least a "prepared" status. The events do not have to be finalized or closed. After running Event Maintenance, confirm that no more pending changes exist by verifying on the BAS Activity page that the BAS Activity grid is empty.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Benefits Administration	All	All

## Task 1-10-2: Preparing COBRA Administration

This step is to be performed by the Benefits Administrator. Run this step if you are using the COBRA Administration module within Base Benefits.

You need to create events for all pending data changes that could trigger COBRA qualification because data conversion cannot convert pending data change "triggers." To create events, perform a COBRA process run. Then review all COBRA Administration messages to ensure that all pending activity has been processed to events. The events do not have to be finalized or closed. After running the COBRA process, confirm that no more pending changes exist by verifying that the COBRA Activity grid is empty on the COBRA Activity page.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 1-11: Making Functional Decisions

This section discusses:

- Understanding Making Functional Decisions
- Assigning Upgrade Default Values
- Assigning Training Employee Record Default Values
- Preparing Workflow Transactions
- Completing eCompensation Transactions
- Completing eDevelopment Transactions
- Mapping Equation Values
- Exporting Upgrade Default Options
- Upgrading Defaults for Service Indicators

## Understanding Making Functional Decisions

In this task, you choose the default values that will be assigned to fields by conversion programs during the upgrade.

## Task 1-11-1: Assigning Upgrade Default Values

This section discusses:

- Understanding Upgrade Default Values
- Opening the Upgrade Defaults Dialog Box
- Assigning the HR 8.01 Default Values
- Assigning Upgrade Defaults for Performance Management
- Mapping Recruiting Solutions Upgrade Fields
- Setting Values for Alternative Overtime Upgrade
- Assigning Schedule Defaults
- Assigning Campus Community Upgrade Default Values
- Assigning Profile Management Options
- Assigning HR Default Values
- Defining the Person Model Conversion Method
- Assigning Student Financials Upgrade Default Values
- Assigning Student Records Upgrade Default Values

### Understanding Upgrade Default Values

This upgrade contains many new fields that need default values. In addition, current values will need to be mapped to new values. In this step, you choose your default values that will be assigned to fields by conversion programs during the upgrade. You will also structure some of the mapping to the way in which you will use the system.

### Opening the Upgrade Defaults Dialog Box

To open the Upgrade Defaults dialog box:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Use, Define Upgrade Defaults.
3. Follow the instructions in the remainder of this task to complete the Upgrade Default pages.

### Assigning the HR 8.01 Default Values

In this step, you assign HRMS 8.01 default values. This dialog box supplies defaults for several conversion processes. Follow the instructions below to assign the values.



SetID:  

Currency Code:  

Define Upg Options page

To assign the HRMS 8.01 default values:

1. Select a valid setID value to be used to automatically populate the SetID field throughout the conversion process. The setID value is used to update the EMPL\_CTG records for labor agreements.
2. Select a valid currency code value to be used to automatically populate the Currency Code field throughout the conversion process. The currency code is used to update the Spanish scheme tables.

## Assigning Upgrade Defaults for Performance Management

In this step, you assign the default values for Performance Management. Complete the information on the Performance Management tab using the instructions below.

In the new PeopleSoft release, the structure of the review rating criteria used in Performance Management and Salary Planning has changed. Specifically, the Review Band series defined in the Merit Scale Table are being consolidated into the Rating Model structure. Under this structure, the Review Band code has changed from four characters to a single character. Therefore, any existing Review Bands with keys longer than a single character must be renamed or mapped to a new code of only a single character (A-Z or 0-9).

**Note.** If there are no Review Band codes that need mapping (either you do not have any Review Bands defined or all codes are already single-characters) then the grid will be disabled and no action is required.

Map Existing Review Bands			View All	First	1-9 of 9	Last
Review Band	New Code	Description				
A	A	Review Band A				
B	B	Review Band B				
S	S	Review Band S				
HXP	<input type="text" value="X"/>	Plan Salaries Review Band Table				
HXA	<input type="text" value="1"/>	Review Band A				
HXB	<input type="text" value="2"/>	Review Band B				
HXC	<input type="text" value="3"/>	Review Band C				
HXD	<input type="text" value="4"/>	Review Band D				
HXE	<input type="text"/>	Review Band E				

Performance Mgmt page

The grid above shows all existing Review Band codes. For any code that is longer than a single character, a data entry field is available for you to define a new single-character code for that Review Band. It is very important that you do not leave any Review Bands with codes longer than a single character. During the upgrade, these codes will be truncated and this could create conflicts with previously defined codes. For example, codes “ABCD,” “XYZ,” and “A123” would all be truncated to “A,” creating a conflict.

**Note.** If you have more than 36 Review Bands defined (although this is highly unlikely) the system will not be able to upgrade them accurately since the number of Review Bands exceeds the number of available codes. In this case, you must “map” the codes that are most important and delete the remaining. To delete excess Review Bands, select Administer Workforce, Plan Salaries (GBL), Setup, Review Band Table.

## Mapping Recruiting Solutions Upgrade Fields

In this step, you assign upgrade values for Recruiting Solutions. Complete the information on the Recruiting tab using the instructions below.

In this PeopleSoft release, several Recruiting Solutions changes occurred. For example, offer amounts are now categorized by offer component types; resume attachments can be stored in the database rather than a file server if desired; recruit statuses and reasons have moved from a translate table to a user-defined table with system-delivered values that can be modified.

To map recruit workforce upgrade options:

1. Select the Recruiting tab.

The screenshot displays the 'Recruiting' tab interface. At the top, there is a section for 'Offer Component' with an empty text box and a 'URL' field containing 'record://HRS\_ATTACHMENTS'. Below this is a 'Status' section with a header bar showing 'First', '1 of 6', and 'Last'. The main area is titled 'Recruitment Area: 1-Job Requisition' and contains a grid with columns 'Job Requisition Status' and 'Code'. The grid lists six rows of status codes: 1 Closed (110), 2 Filled (110), 3 Hold (100), 4 Cancelled (120), 5 Open (010), and 6 Re-Opened (010). Each row has '+' and '-' buttons. Below the grid is another section with a header bar showing 'First', '1 of 1', and 'Last'. This section contains a table with columns 'Status Code' and 'Reason', showing one row with an empty status code and reason, and '+' and '-' buttons.

Recruiting tab

2. Enter a value in the Offer Component field.

The Recruit Workforce Upgrade moves all existing offers into the HRS\_OFF\_DTL record and uses this value as the default OFFER\_COMPONENT field. When the upgrade is finished, create an offer component that is equivalent to the value entered here.

3. For conversion purposes, enter the actual URL you will be using for centralized resume location in the URL field.

---

**Note.** The URL field is only pertinent if you are using Resume Processing. In this PeopleSoft release, resume attachments will be stored in one location regardless of whether the resume came from Resume Processing, Apply Online, or was attached in Applicant Data. This location is specified in Set Up HRMS, Product Related, Recruiting, Attachment Types for the specific resume type. If you choose, you may store your resume attachments in the database instead of in a file server by specifying `<record://HRS_ATTACHMENTS>` in the URL field.

---



---

**Note.** The remaining Recruiting options only need consideration if you have added translate values to the following fields: APP\_STATUS, RESPONSE\_CD, RESPONSE\_OFFER, JOB\_REQ\_STATUS, RECOMMENDATION, DISPOSITION\_REASON, or OFFER\_REJECT\_RSN. If you have not added any additional values to these fields, you have completed the Recruiting upgrade page and you can move on to the next step.

---

4. Scroll to the appropriate Recruitment Area where you have added translate values: 1 (Job Requisition) – JOB\_REQ\_STATUS, 2 (Applicant) – APP\_STATUS for APPLICANT status, 3 (Position Applied For) – APP\_FOR\_STATUS for POSN\_APPLIEDFOR table, 4 (Route) – RESPONSE\_CD, 5 (Interview) – RECOMMENDATION, or 6 (Offer) – RESPONSE\_OFFER.

Each Recruitment Area corresponds to one of the translate values that is being converted.

5. Insert a row for each translate value that you have entered.

---

**Note.** Do not delete any of the rows that currently exist.

---

6. Map the status code that most closely represents the translate value you have added.

Refer to the appendix “Reviewing HRMS Changes” for a list of values delivered as system data.

If a matching status code does not exist in the list below, you may add a new status. Following the upgrade, add this new Status/Reasons component under Set Up HRMS, Product Related, Recruiting, Status Reason.

For Recruitment area 4 (Route), you may also specify a status reason that may correspond to a rejected status.

If you have added any translate values to the DISPOSITION\_REASON or OFFER\_REJECT\_RSN fields, you will need to fill in the second grid on this page. The upgrade conversion only looks at Recruitment Areas 3 (Position Applied For) and 6 (Offer). You may ignore the other Recruitment areas. Insert a row for each translate value that you have entered. Do not delete any of the rows that currently exist. Following the upgrade, add this new status code in the Status/Reasons component under Set Up HRMS, Product Related, Recruiting, Status Effects.

Previous to this PeopleSoft release, the Federal installation of Recruiting delivered 28 additional DISPOSITION\_REASON values. As a result of the Status Enhancement feature, the number of delivered DISPOSITION\_REASON (now STATUS\_REASON) values was significantly reduced. The values that are no longer delivered are listed below:

- CE: Career/Career Conditional Empl
- CR: Communication Returned Unclaimed

- DA: Declined Agency
- DD: Declined Until a Later Date
- DE: Declined Examination Program
- DG: Declined Grade
- DL: Declined Location
- DP: Declined for Posn Certified
- DX: Declined Further Consideration
- DZ: Declined for Other Reasons
- FR: Failed to Reply
- NC: Appt by Non-Competitive Action
- NN: Not Selected - Not Contacted
- RM: Removed from Certif - Medical
- RQ: Removed from Certif - Not Qual
- RS: Removed frm Cert - Suitability
- TE: Temp (or Indefinite) Employee

---

**Note.** If these values are currently used in your system, you must insert a row into the Rejected Reason grid for each value used and match the translate value with an appropriate status code.

---

## Setting Values for Alternative Overtime Upgrade


The Alternative Overtime upgrade is for sites with North American Payroll installed. If you do not have this product installed, clear the Perform Upgrade check box on the Alternative Overtime page. If you have this product, you need to review the Alternative Overtime feature in the new release to determine whether you want the upgrade steps performed.

In this step, you specify whether you want to do the Alternative Overtime Upgrade. If so, enter values in the Effective Date for Upgrade, SetID, and State Code fields.

To set the fields required for Alternative Overtime functionality:




1. Select the Alt Overtime tab.

























☒ **Perform Upgrade**

**Effective Date for Upgrade:**  

View All First ◀ 1 of 1 ▶ Last

**SetID**

State	Description		
AK	 Alaska		
CA	 California		
CO	 Colorado		
KS	 Kansas		
KY	 Kentucky		
NV	 Nevada		
PR	 Puerto Rico		
RI	 Rhode Island		

Alt Overtime tab

- Select the Perform Upgrade check box if you want to perform the Alternative Overtime Upgrade.  
If you select this check box, you must enter all other fields on this page.
- Enter a date in the Effective Date for Upgrade field. Do not confuse this date with the effective date on the Alternative Overtime State table, which will be set to 1900/01/01 by default. The date specified here will be used for inserting new effective dated rows into the Job table with a new FLSA status of *Alt OT* for all employees who meet all of the following criteria:
  - Currently have FLSA Status = Nonexempt
  - HR Status is not equal to Terminated
  - Work in a state specified on the Alt Overtime upgrade defaults page
- Enter a setID in the SetID field.  
The setID specified here will be used for inserting rows into the Alternative Overtime State table. Each setID will be associated with all states listed below.
- Add or delete states by entering or deleting values in the State Code field, as needed. The list of states is pre-populated with the states that require Alternative Overtime processing.  
All states will be associated with each setID listed above.

## Assigning Schedule Defaults

In this step, you populate the new setID and Schedule Group fields that have been added to scheduling objects.

To map setID and schedule groups:

1. Select the Schedules tab.

## Upgrade Schedule

To use one SETID for all Schedule Definitions (and their related shifts and workdays), enter a Default SETID and create a Default Schedule Group and description.

**Schedule Defaults**

**Schedule SetID:** 
**Schedule Group:** 
**Description:**

To map individual schedules, create different Schedule Groups, then associate a Schedule ID to only one Schedule Group.

**Additional Schedule Groups**

First 1-2 of 2 Last

*Schedule Group	*SetID	*Description		
<input type="text" value="GXSCHEGR"/>	<input type="text" value="GXSID"/>	<input type="text" value="GP Core Test Schedule Group"/>	<input type="button" value="+"/>	<input type="button" value="-"/>
<input type="text" value="MODELGRP"/>	<input type="text" value="MODEL"/>	<input type="text" value="Model Schedule Group"/>	<input type="button" value="+"/>	<input type="button" value="-"/>

**Schedule Mapping**

First 1-8 of 8 Last

*Schedule ID	*Schedule Group		
<input type="text" value="GXHOL1"/>	<input type="text" value="GXSCHEGR"/>	<input type="button" value="+"/>	<input type="button" value="-"/>
<input type="text" value="GXWRKSCH"/>	<input type="text" value="GXSCHEGR"/>	<input type="button" value="+"/>	<input type="button" value="-"/>
<input type="text" value="KUSCHDFE1"/>	<input type="text" value="MODELGRP"/>	<input type="button" value="+"/>	<input type="button" value="-"/>
<input type="text" value="KUSCHDFE2"/>	<input type="text" value="MODELGRP"/>	<input type="button" value="+"/>	<input type="button" value="-"/>

Schedules tab

2. If only a single schedule group and setID will be used, complete the following fields in the Schedule Defaults section to set up the upgrade for the schedule objects:
  - Enter a setID in the Schedule SetID field.  
The setID specified here will be used to populate the SetID field on all scheduling objects (Shift, Workdays, Schedule Definitions, and Schedule Overrides). This setID must already be defined.
  - Enter a value in the Schedule Group field that is to be associated with the above setID.  
The value cannot be greater than 10 characters in length. The upgrade process creates this schedule group.
  - Enter a description of the schedule group in the Description field (no more than 20 characters in length).
3. If schedules will be split up and assigned to more than one schedule group or setID, complete the following fields in the Additional Schedule Groups section, entering a row in this section for each additional schedule group to be created during the upgrade:
  - Enter a value in the Schedule Group field that will be associated with the above setID (no more than 10 characters in length).  
This schedule group will be created by the upgrade process.

- Enter a setID in the SetID field.

The setID specified here will be used to populate the setID field for the associated schedule group when it is created during the upgrade. Each schedule group must be associated with a setID. This setID must already be defined.

- Enter a description of the schedule group in the Description field (no more than 30 characters in length). Each schedule group must be given a description.

4. If you have schedules that will *not* use the default schedule group and setID, complete the fields in the Schedule Mapping section to define the associated schedule group.

Each schedule can only be associated with one schedule group. The setID assigned to the schedule will be the setID of the selected schedule group.

- Enter a value in the Schedule ID field.

The schedule ID will be associated with the schedule group.

- Enter a value for the schedule group that is to be associated with the above schedule ID.

The value must be defined in the Additional Schedule Groups section.

---

**Note.** The Schedule Group field on employee Schedule Assignments, Time and Labor Workgroups, and GP Paygroup will be populated using the schedule group associated with the schedule to which they are assigned.

The Alternate SetID and Schedule Group fields are populated wherever an alternate schedule ID is present, using the same mappings that are used for the primary schedule.

---

For additional information about effective date and schedule definition, as well as schedule ID, see the reference below.

See Appendix: “Reviewing HRMS Changes,” Reviewing Schedule Defaults.

## Assigning Campus Community Upgrade Default Values

In this step, you define the default values used in the Campus Solutions upgrade.

The screenshot shows a web interface with two fields. The first field is labeled "Regulatory Region:" and contains a text box with "USA" and a magnifying glass icon, followed by the text "United States". The second field is labeled "Phone Type:" and contains a dropdown menu with "Campus" selected.

Campus Community page

The regulatory region value is used to populate the new Regulatory Region field used by the Core Human Resources product on the DIVERS\_ETHNIC record. The DIVERS\_ETHNIC record is now used in place of the DIVERSITY record to store one or many ethnic group code values for a person. The Campus Community upgrade retains ETHNICITY\_DTL information entered in Student Administration 8.0 utilizing both the Core Human Resources DIVERS\_ETHNIC record and the ETHNICITY\_DTL record. In the new PeopleSoft release, the ETHNICITY\_DTL record has been altered to be an extension of the DIVERS\_ETHNIC record. If a related DIVERS\_ETHNIC record is not found for existing ETHNICITY\_DTL records, one is inserted and the related ETHNICITY\_DTL record is generated with like key values.

The phone type value becomes the preferred phone type for anyone who has not chosen a preferred phone type in the PeopleSoft release. If an ID does not have a record for this phone type and does not have a preferred phone type selected, the preferred phone type will be set based on the phone types in the following sequence: Main, Business, Home. If there is still no preferred phone type selected, the first phone number found will be set as the preferred phone type.

## Assigning Profile Management Options

In this step, you determine the default values to be used in data conversion for the Profile Management upgrade. If no values are supplied, the defaults described below will be applied by the conversion program.

To set Profile Management options:

1. Select the Profile Management tab.

<b>Profile ID Length:</b>	<input type="text" value="8"/>
<b>Last Profile ID Used:</b>	<input type="text" value="100000"/>
<b>Last Item Key ID Used:</b>	<input type="text" value="1000"/>

Profile Management Tab

2. In the Profile ID Length field, enter the number of characters that you will need for the new profile ID.  
The default value is 6. During conversion, new auto-generated numeric profile IDs will be created for Person Profiles, Clusters, Roles, and Job Profiles. Once in production, new profiles will be assigned an auto-generated number (stored in a character field). The default value supplied is intended to accommodate all profile IDs generated during conversion as well as a vast supply of IDs once in production. The default value also accommodates characters for a prefix to be used when creating IDs in production, if designated.  
If there is concern that your company will be upgrading more than 899,999 profiles, this would be a good time to increase the profile ID length, though the conversion will not error if profile IDs longer than the profile ID length specified are created. This value can be increased after the upgrade if special needs should be accommodated for profiles created while in production.
3. Set the Last Profile ID Used field to the starting number you want to use.  
In the new release, Profile Management uses profile ID in a numeric sequence. The default is to start the sequence at 100,000, thus the first profile that will be created during the upgrade will have a profile ID of 100,001. You may want to start at a number other than 100,000. Set this field to the starting number you wish to use; however, be aware of the profile ID length, ensuring that the starting number you enter can accommodate profiles being converted and still remain within the profile ID length. In most cases there is



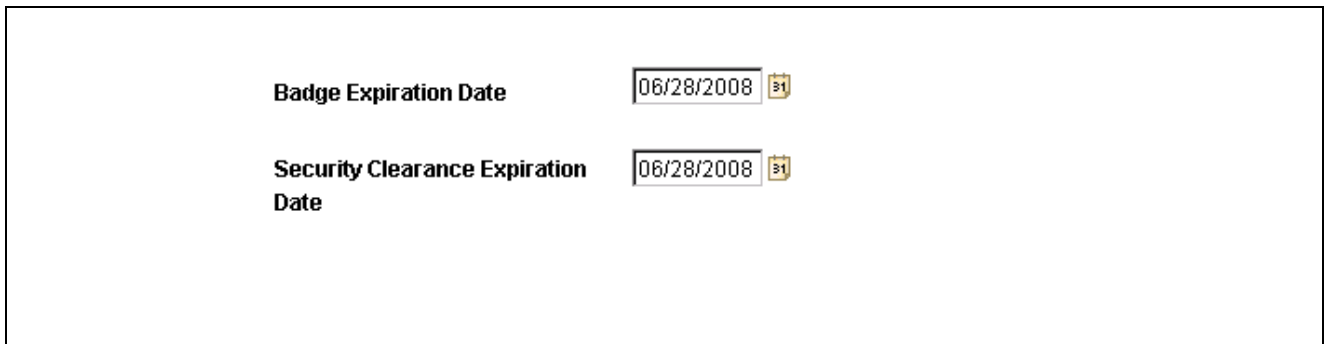
no need to change this number. A valid case would be if a custom process were used import profiles from an alternative source having unique profile IDs that need to be protected.

4. Set the Last Item Key ID Used field to the starting number you want to use.

In the new release, Profile Management will assign a unique numeric ID to all items in a profile. The default is to start the sequence at 0; therefore, the first profile item created will have a starting number of 1. If you want to start with another number, set this field to that number. In most cases there is no need to change this number. A valid case would be if a custom process were used to import profiles from an alternative source having a series of IDs that need to be reserved.

## Assigning HR Default Values

In the new release, the EXPIRATION\_DATE field is added to the BADGE\_TBL record. The EXPIRATION\_DATE field also exists on the new Employee Security Clearance record, HR\_EE\_SEC\_CLR. Application errors may occur when the expiration date is before the effective date. To avoid potential errors, specify values for the Badge Expiration Date and Security Clearance Expiration Date fields on the Define HR Defaults page.

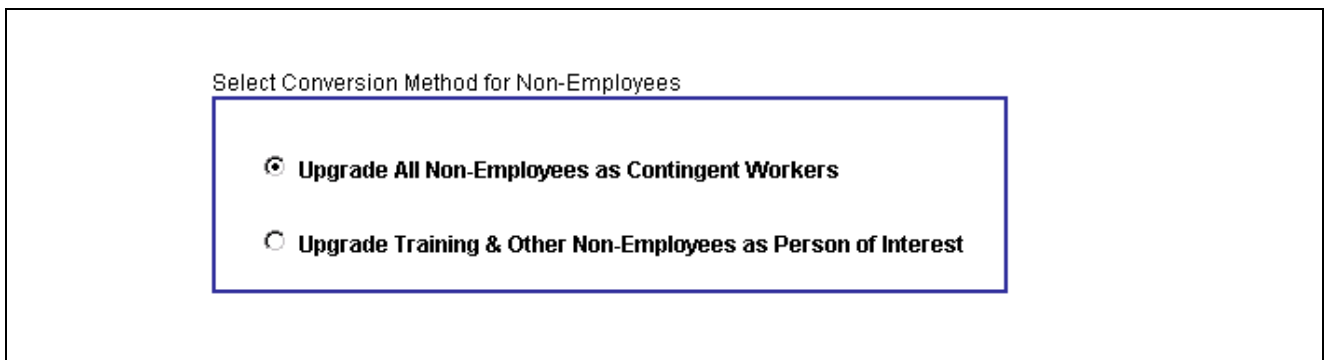


The screenshot shows two date input fields on a white background. The first field is labeled "Badge Expiration Date" and contains the date "06/28/2008" with a calendar icon to its right. The second field is labeled "Security Clearance Expiration Date" and also contains the date "06/28/2008" with a calendar icon to its right.

Define HR Defaults page

## Defining the Person Model Conversion Method

In this step, you define the person model conversion method. Review the appendix “Understanding Person Model Changes.” It discusses features in the new release and the default method for converting your data. If you are satisfied with the design of the conversion, you can skip the Person Model Defaults page. If you prefer to have your Non-Employees converted as Persons of Interest instead of Contingent Workers, select the conversion method option “Upgrade Training & Other Non-Employees as Person of Interest” on this page.



The screenshot shows a section titled "Select Conversion Method for Non-Employees" with a blue border. Inside, there are two radio button options. The first option, "Upgrade All Non-Employees as Contingent Workers", is selected with a filled radio button. The second option, "Upgrade Training & Other Non-Employees as Person of Interest", is unselected with an empty radio button.

Person Model Defaults page

Changing the default conversion method will result in your data converting as follows:

PER_TYPE	Upgrade Conversion Chart Descr	PER_ORG	POI_TYPE	JOB
O	Other Non-Employee	POI	'00011'	Yes Other Payee
T	Training Non-Employee	POI	'00011'	Yes Other Payee
T	Training Non-Employee Instructors	POI	'00011'	Yes Other Payee

Chart 2 - Upgrade Conversion Chart for PER\_STATUS and PER\_TYPE

**Note.** This chart only outlines the differences from what is described in the appendix “Understanding Person Model Changes.”

## Assigning Student Financials Upgrade Default Values

In this step, you define the default values used in the Student Financials Global Invoicing conversion process.

**Note.** Skip this step if you do not intend to use the Student Financials Global Invoicing feature.

To assign Student Financials upgrade defaults:

1. Select the Student Financials tab.
2. Select the Use Global Invoicing check box.
3. Enter a default value in the Pre-conversion Invoice ID field.

☒ Use Global Invoicing

Pre-conversion Invoice ID:

Student Financials tab

When the Use Global Invoicing check box is selected, the Student Financials upgrade pre-populates the table SSF\_ITEM\_INV using the pre-conversion invoice ID. This table is only used to support Global Invoicing. The Student Financials Global Invoicing feature uses this new table to support Credit Invoicing and the payment by invoice ID.

## Assigning Student Records Upgrade Default Values

In this step you define the default values used in the Student Records upgrade.

The option to round or truncate GPA fields in Student Records (PS\_STDNT\_CAR\_TERM) is now available in PeopleSoft 9.0. In previous releases, this was done by hard coding in COBOL and Peoplecode programs. Because new GPA fields are added in Student Records (PS\_STDNT\_CAR\_TERM) in PeopleSoft 9.0, the upgrade process requires that you identify how GPA rounding or truncating is done at your installation. This will ensure that the new GPA fields are rounded or truncated the same way as the old GPA fields. Note that the upgrade is only going to calculate GPA values for the new fields.

To assign Student Records defaults:

1. Select the Student Records tab.

Student Records tab

2. Select the appropriate GPA rounding option from the GPA Rounding/Truncating Option drop down list box.

The rounding/truncation option value you define in this step must be consistent with how you have rounded or truncated in the past. For example, if you have been rounding the GPA fields using 2 decimal positions, you must select “Round to 2 decimal places” in this option.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-11-2: Assigning Training Employee Record Default Values

In the new release, the EMPL\_RCD field is added to the TRAINING and TRN\_INSTRCT\_TBL records as a key. In this step, you assign the proper employee record having training association for trainees and instructors with multiple employee job records.

Define Training Employee Record page

To assign training employee record values:

1. Select Define Business Rules, Define Upgrade Options, Use, Define Upgrade Defaults, Employee Training.
2. Click the Load Upg Training Default Page button.

This displays all trainees and instructors with multiple employee records in PS\_JOB.

3. Select the “Upgrade?” check box next to the EMPL\_RCD value to populate the TRAINING and TRN\_INSTRUCT\_TBL records during the conversion process.

---

**Note.** Failure to select the EMPL\_RCD to upgrade will result in default values of zero.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-11-3: Preparing Workflow Transactions

In this step, you verify that all of your workflow transactions are complete. Complete this step if you are using the eCompensation or eDevelopment modules.

To verify that all transactions are complete:

1. Select PeopleTools, Worklist, Use, Worklist.
2. Verify that all transactions have the status *Done*.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	eCompensation eDevelopment	All	All

## Task 1-11-4: Completing eCompensation Transactions

The salary change transactions in the new PeopleSoft release replace both the Employee Salary Change and Group Salary Change without Budget transactions. The table structure supporting the new transaction has changed to support new functionality, including workflow notifications and the ability to update core tables via automatic database updates. You should finish all transactions that are in process before running the conversion program.

- Employee Salary Data. The Change Employee Salary Detail data is stored as an email in one field, but the conversion process does not parse these email entries. The data is placed into the new transaction tables.
- Group Salary Data. The data saved for later transactions and group calculation amounts will be lost during upgrade because this functionality is not supported in the new PeopleSoft release.

To verify that all transactions are complete:

1. Select Self Service, Manager, Tasks, Employee Salary Change Status.
2. Verify that there are no active employee salary change requests. If there are salary change requests, you will need to work with the approvers to approve or deny any changes.
3. Select Self Service, Manager, Tasks, Group Salary Change Status.

4. Verify that there are no groups without allocated budget. If there are groups without allocated budgets, you will need to work with those who approve or deny the group requests.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	eCompensation eDevelopment	All	All

## Task 1-11-5: Completing eDevelopment Transactions

The table structure supporting the new transaction has changed to support new functionality, including workflow notifications and the ability to update core tables via automatic database updates. It is recommended you finish all transactions in progress before running the conversion process. The records PS\_HR\_ENROLL\_DAT and PS\_HR\_ENROLL\_STA will be deprecated in the future.

To verify that all transactions are complete:

1. Select Self Service, Manager, Tasks, Approve Training.
2. Verify that there are no active training requests.

If there are training requests, you will need to work with those who approve or deny the requests.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	eCompensation eDevelopment	All	All

## Task 1-11-6: Mapping Equation Values

This section discusses:

- Understanding Mapping Equation Values
- Opening the Equations Upgrade Defaults dialog box
- Updating Financial Aid Equations
- Updating Student Financials Equations

### Understanding Mapping Equation Values

The Student Financials and Financial Aid Equations in your new release are combined. If you have equations using the same name in both systems, they will not be combined. You must give the equations new names to allow the data to be combined. Also, old demo equations will not migrate into the new release. It is recommended that you use the new release version of the demo equation.

In this step, you rename current equations and determine which equations will not be upgraded. If you keep one of the demo equations, you must rename it so that it is not dropped.

## Opening the Equations Upgrade Defaults dialog box

To open the Equations Upgrade Defaults dialog box:

1. From your web browser, sign in to the Copy of Production database.
2. Select Define Business Rules, Define Upgrade Options, Use, Equations Upgrade Defaults.
3. Update the appropriate pages, following the instructions in the steps below.

**Note.** The Aid Fields and Financial Fields pages are reference only. They display the field transformation that will be used during data conversion. The data for these pages is delivered with the upgrade and should not be changed.

## Updating Financial Aid Equations

Select the Financial Aid tab to review the demo equations and update your equation names. The Equation Name Old column contains the name in your current production Financial Aid equations. The Equation Name New column will be blank if the equation will be upgraded. If you enter a new name here, the equation will be renamed during the upgrade. The message column will display a message if the equation is demo data and will not be upgraded.

**Note.** Even if you do not make any changes to data on this page, you *must* click the Save button before leaving the page. Otherwise, no equations will be converted. This is because the table is automatically populated with a list of existing equations when you open the page.

Financial Aid    Aid Fields    Student Financials    Financial Fields			
View All    First 1-13 of 77 Last			
Equation Name Old	Equation Name New	Message Text	
1 ADJ_P_CONT	FAPDADJPCON	Demo Data Name, Will Not Convert. (14120,4)	+ -
2 ANNUAL_LOAD	FAPDANNLOAD	Demo Data Name, Will Not Convert. (14120,4)	+ -
3 ANNUAL_LOAD2	FAPDANNLOAD	Demo Data Name, Will Not Convert. (14120,4)	+ -
4 ANNUAL_LOAD3	FAPDANNLOAD	Demo Data Name, Will Not Convert. (14120,4)	+ -
5 APOLLO	FAPDAPOLLO	Demo Data Name, Will Not Convert. (14120,4)	+ -
6 APP_DATE	FAPDAPP_DATE	Demo Data Name, Will Not Convert. (14120,4)	+ -
7 BJF TEST DEP	FAPDBJFTSTDP	Demo Data Name, Will Not Convert. (14120,4)	+ -
8 CALC-IF-AGI	FAPDCLCIFAGI	Demo Data Name, Will Not Convert. (14120,4)	+ -
9 CALC-TLU-EFC	FAPDCLCTLEFC	Demo Data Name, Will Not Convert. (14120,4)	+ -
10 DEPENDENT	FAPDDEPENDN	Demo Data Name, Will Not Convert. (14120,4)	+ -
11 DEPENDENTS	FAPDDEPNNT	Demo Data Name, Will Not Convert. (14120,4)	+ -
12 DL_STATE	FAPDDL_STATE	Demo Data Name, Will Not Convert. (14120,4)	+ -
13 DOB_PRIOR	FAPDDOBPRIOF	Demo Data Name, Will Not Convert. (14120,4)	+ -

Financial Aid tab

## Updating Student Financials Equations

Select the Student Financials tab to review the demo equations and update your equation names. The Equation Name Old column contains the name in your current production Student Financials equations. The Equation Name New column will be blank if the equation will be upgraded. If you enter a new name here, the equation will be renamed during the upgrade. The message column will display a message if the equation is demo data and will not be upgraded.

**Note.** Even if you do not make any changes to data on this page, you *must* click the Save button before leaving the page. Otherwise, no equations will be converted. This is because the table is automatically populated with a list of existing equations when you open the page.

Financial Aid    Aid Fields <b>Student Financials</b> Financial Fields			
View All    First 1-7 of 7 Last			
Equation Name Old	Equation Name New	Message Text	
1 CASESTUDYFEE	SFTDCSSTDYFE	Demo Data Name, Will Not Convert. (14120,4)	+ -
2 MDLABORATORY	SFTDMDLABRA	Demo Data Name, Will Not Convert. (14120,4)	+ -
3 MDSCHTUITION	SFTDMDSCTUIT	Demo Data Name, Will Not Convert. (14120,4)	+ -
4 MDSPECIALITY	SFTDMDSPECL	Demo Data Name, Will Not Convert. (14120,4)	+ -
5 MDTESTSCORES	SFTDMDTESTS	Demo Data Name, Will Not Convert. (14120,4)	+ -
6 RESEARCHFEES	SFTDRSRCHFEE	Demo Data Name, Will Not Convert. (14120,4)	+ -
7 UGRAD	SFTDUGRAD	Demo Data Name, Will Not Convert. (14120,4)	+ -

Student Financials tab

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-11-7: Exporting Upgrade Default Options

In this step, you export the upgrade default options defined in the previous steps for use during Move to Production.

The script name for your upgrade is:

PUHCHRW01E.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-11-8: Upgrading Defaults for Service Indicators

This section discusses:

- Reviewing the Service Indicator/Impact Report
- Upgrading Defaults for Service Impacts
- Upgrading Defaults for Service Indicator Codes
- Upgrading Service Indicator Code and Impact Associations

### Reviewing the Service Indicator/Impact Report

A new attribute, Basis of Impact, has been added to the Service Impact definition. Basis of Impact indicates whether a given impact takes effect based on date, term, or both. This new attribute, combined with the new End Date and End Term fields and the standardization of Term field usage in service indicator assignments, requires that you assess current Service Impact and Service Indicator Code associations to determine the proper upgrade strategy.

See Appendix: “Understanding Service Indicator Changes.”

To assist with this assessment, the Pre-Upgrade Service Indicator report, PUCCR10, is provided. This report lists the impacts defined in your implementation and the Service Indicator codes with which those impacts are associated. To determine the appropriate upgrade strategy, you will need to apply your knowledge of your institutional usage of the impacts and codes.

To run the Service Indicator/Impact report:

1. Select Define Business Rules, Define Upgrade Options, Report, Pre-Upgrade Service Indicator.
2. On the run control page, click Run to run the report.
3. Review the Pre-Upgrade Service Indicator report.

### Upgrading Defaults for Service Impacts

For each defined impact, you must select a basis of impact (date, term, or both) based on your analysis of how the impacts are used. You must select the basis of impact for all impacts for all institutions for which you have Service Impacts defined.

To upgrade Service Impact defaults:

1. Select Define Business Rules, Define Upgrade Options, Use, Upgrade Service Indicators.



[Home](#) > [Define Business Rules](#) > [Define Upgrade Options](#) > [Use](#) > **Upgrade Service Indicators**

Upgrade Service Impact   Upgrade Service Indicator Code

**Service Impacts** View All First 1 of 1 Last

Institution: PSUNV

**Impacts** View All First 1-6 of 22 Last

Impact	Description	Term	Date
ACCES	Disabled Access Privilege	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AENR	Allow drop only; no add actvty	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
AWARD	Award Financial Aid	<input checked="" type="checkbox"/>	<input type="checkbox"/>
BILL	Receive a Bill	<input type="checkbox"/>	<input checked="" type="checkbox"/>
BILLS	No Bill	<input type="checkbox"/>	<input checked="" type="checkbox"/>
CASH	Cash Check	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Upgrade Service Impact tab

- On the Upgrade Service Impact tab, enter a value in the Institution field.
- For each impact listed in the Impacts section, select either the Term or Date check box, or both, to indicate the basis of impact.
- Repeat steps 2 and 3 for each institution for which you have Service Impacts defined.

## Upgrading Defaults for Service Indicator Codes

To accommodate the exposure of Service Indicator functionality to external organizations in PeopleSoft 9, a new attribute has been added to the Service Indicator Code definition that allows you to decide whether to use a Service Indicator code with Persons, or Organizations, or both. If a Service Indicator code is not marked for usage with an entity type, it will not be available in the Service Indicator Code prompt for that entity type.

For example, if Service Indicator Code SF9 has the *Person* check box selected, but not the *Organization* check box, then when you add a new Service Indicator for an organization, SF9 will not appear in the prompt as a valid Service Indicator that can be assigned.

You must define your Service Indicator Code upgrade defaults for each institution for which you have Service Indicator codes defined.

To upgrade Service Indicator code defaults:

- Select Define Business Rules, Define Upgrade Options, Use, Upgrade Service Indicators.

Upgrade Service Impact   Upgrade Service Indicator Code

**Service Indicator Codes** View All First 1 of 1 Last

Institution: PSUNV

**Codes** View All First 1-8 of 46 Last

Service Indicator Cd	Description	Person	Organization
ALL	All Services Hold	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
ANO	Anonymous	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B01	No Bill Created	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
BIL	Billing Indicator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
C01	Conference Guest	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CMP	Campaign 2000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DEP	Enrollment Deposit	<input checked="" type="checkbox"/>	<input type="checkbox"/>
DNM	Do Not Mail	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Upgrade Service Indicator Code tab

- On the Upgrade Service Indicator Code tab, enter a value in the Institution field.
- For each Service Indicator code listed in the Codes section, select either the Person or Organization check box, or both.
- Repeat steps 2 and 3 for each institution for which you have Service Indicator codes defined.

**Note.** If a Service Indicator Code does not have either the Persons or Organizations check box selected, it will not show up in either prompt.

## Upgrading Service Indicator Code and Impact Associations

As noted above, because of changes to the Service Indicator functionality intended to make interpretation of values consistent across products, you will need to make decisions about Service Indicator code and impact associations and other Service Indicator values in order to ensure that the functional intent of upgraded assigned Service Indicators remains the same in PeopleSoft 9 as it was in the source release.

**Note.** You must define upgrade defaults for Service Impacts and Service Indicator codes before you can upgrade Service Indicator code and impact associations.

You must define the Service Indicator code and impact association upgrade defaults for each institution for which you have Service Indicator codes defined. For each Service Indicator code for a selected institution, you must make choices in three areas:

- Service code and impact associations
- Upgrade target value for End Term
- Upgrade value for use of Active Term = 9999

To upgrade Service Indicator and impact associations:

- Select Define Business Rules, Define Upgrade Options, Use, Upgrade SI Code/Impacts.
- On the Search page, select the institution for which you want to define Service Indicator code and impact upgrade defaults.

**Upgrade Service Indicator Code/Impacts**

Institution: PSUNV

**Service Impacts** View All First 47 of 54 Last

Service: U03 Map: ☐ Direct ☒ Split

Indicator Cd: Effective Date: 12/31/2001

Impacts	Basis of Impact
BILL	<input type="checkbox"/> Term <input checked="" type="checkbox"/> Date
OOT	<input checked="" type="checkbox"/> Term <input type="checkbox"/> Date
PAYMN	<input type="checkbox"/> Term <input checked="" type="checkbox"/> Date

**Map To** View All First 1 of 1 Last

Code: End Term Equals: Convert '9999' to: 9999

Description: ☐ Blank ☐ Start Term

Impact	Description
<input type="text"/>	<input type="text"/>

Upgrade Service Indicator Code/Impacts page

For each effective dated row for a source Service Indicator code, the impacts associated with that code are displayed. In addition, each impact displays the basis of impact that was selected in the step Upgrading Defaults for Service Impacts. This will help you identify “mixed usage” cases.

- To associate a Service Indicator code with the impacts, under Map select the method of mapping the code, either Direct or Split.

Direct mapping will copy the code definition to the upgrade target without changes. Split mapping provides an opportunity to reassociate impacts with codes (existing or new), depending on your institutional usage, in order to retain the original assignment intention based on the new structures. If a Service Indicator code does not have any associated impacts, the Map option will default to Direct and will not be active.

**Note.** If you add new Service Indicator codes by splitting codes, the new Service Indicators (Institution, Code, and Reason) and their security settings (Place and Release) will be added to each relevant user’s Service Indicator security profile in a manner identical to the source row.

- For End Term Equals, select the Start Term option to populate the new End Term field with the value from Start Term, or select the Blank option to leave the End Term field blank.

**Note.** For details about the historical usage of Active Term and new validity period logic in PeopleSoft 9, as well as examples of upgrade scenarios, see Appendix: “Understanding Service Indicator Changes.”

- For Convert '9999' to, select whether to convert the Active Term value of 9999 to a Start Term value of 0000 or 9999.

**Note.** For details about the historical usage of the Active Term value of 9999 and examples of upgrade scenarios, see Appendix: “Understanding Service Indicator Changes.”

- Repeat steps 3 through 5 for each Service Indicator code for the selected institution.
- Repeat steps 2 through 6 for each institution for which you have Service Indicator codes defined.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	All	All

## Task 1-12: Reporting Row Count for Tables

You may find it helpful to run a report that identifies any table without rows, that is, any table not used in your production database. This information can help you determine whether to accept a change from the New Release Demo database. The UPGCOUNT process reports the row counts of all PeopleSoft tables in your database. You can find the resulting report (UPGCOUNT.LIS) in the TEMP directory specific to your machine.

If you use PeopleTools 8.13 or earlier, use SQR from your current (old) release of PeopleSoft to run the following script against your Copy of Production:

```
\PS_HOME\SQR\UPGCOUNT.SQR
```

In PeopleTools 8.14, UPGCOUNT was converted to an Application Engine program. If your Copy of Production is using PeopleTools 8.14 or later, run UPGCOUNT as an Application Engine program. Using your current (old) release of PeopleSoft, run the Application Engine program UPGCOUNT from the command line on your client workstation against the Target database. The command line command is as follows:

```
\PS_HOME\bin\client\winx86\psae -CT dbtype -CS server -CD dbname -CO oprid -CP =>
oprpswd -R UPG -AI UPGCOUNT
```

**Note.** The -CS option applies to Informix and Sybase installations only.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-13: Preparing Your Database

This section discusses:

- Understanding Database Preparation
- Verify Database Integrity
- Clean the PSOBJCHNG Table
- Shrink Images
- Purge Message Queues

## Understanding Database Preparation

In this task, you perform a variety of steps in preparation for the PeopleTools upgrade. These steps prevent errors in tasks later in the upgrade.

### Task 1-13-1: Verify Database Integrity

The DBCC CHECKDB command is run to perform a database consistency check on your Copy of Production database. A database consistency check ensures that your database platform environment is clean and minimizes any potential upgrade errors due to possible database corruption.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server Sybase	All

### Task 1-13-2: Clean the PSOBJCHNG Table

This step deletes all data stored in the PSOBJCHNG table, which contains all renamed records and fields. The data stored in the PSOBJCHNG table must be deleted before starting your upgrade. The build process looks in this table when running alter renames. Change Assistant will execute the following SQL:

```
DELETE FROM PSOBJCHNG
```

---

**Note.** Move to Production: If you rename records or fields later in your upgrade, you should expect to see rows in the PSOBJCHNG table at the end of the upgrade pass. During the Move to Production these rows will be copied from your old Copy of Production to your new Copy of Production. Thus, this step is not necessary during Move to Production.

---

See “Apply PeopleTools Changes,” Renaming PeopleTools Records.

See “Apply Application Changes,” Modifying the Database Structure.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 1-13-3: Shrink Images

If you have customized images stored in your database, you may need to shrink these images before updating PeopleTools system tables later in the upgrade. Large image fields could cause that step to fail because it is not possible to bind long raw data that is longer than 32 KB.

To shrink images:

1. Launch Configuration Manager and select the Profile tab.
2. Select the profile for the upgrade database and click Edit.

3. Select the Common tab.
4. Select the Convert and Shrink Images to Platform Limit option.
5. Click OK.

---

**Note.** If you re-shrink images, select Don't Convert, but Shrink Images to Image Size Limit. Specify the number of bytes for the image size limit.

---

6. Launch Application Designer.
7. Select Tools, Upgrade, Convert Images...
8. Select Convert Static Images in Image Catalog.
9. Click Start to convert or shrink images.
10. Select Tools, Upgrade, Convert Images...
11. Select Convert Dynamic Images for fields. Select the box for all of the fields listed.
12. Click Start to convert or shrink images.

See "Apply PeopleTools Changes," Updating PeopleTools System Tables.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 1-13-4: Purge Message Queues

Ensure that all of your message transactions are complete before starting the upgrade. Message functionality and structure changed in the new release, which will prevent old messages from processing successfully.

This step runs the following Data Mover script (DMS), found in the %PS\_HOME\SCRIPTS directory of your old release codeline, on your Copy of Production database to purge your message queues:

```
APPMSGPURGEALL.DMS
```

---

**Warning!** A script of the same name is found in the codeline of the release to which you are upgrading. Do not use this script; it will not run successfully.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 1-14: Backing Up Demo Databases

This section discusses:

- Backing Up Copy of Current Demo
- Backing Up New Release Demo

### Task 1-14-1: Backing Up Copy of Current Demo

Back up your Copy of Current Demo database now. This upgrade requires you to run scripts on this database. Before the upgrade starts, you need to take a backup of this environment to preserve your Oracle-delivered demo implementation.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Copy of Current Demo	Initial	All	All	All

### Task 1-14-2: Backing Up New Release Demo

Back up your New Release Demo database now. This upgrade requires you to run scripts on this database. Before the upgrade starts, you need to take a backup of this environment to preserve your Oracle-delivered demo implementation.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

---

## Task 1-15: Renaming Records and Fields

This section discusses:

- Understanding Renaming Records and Fields
- Running the RNHCUPG01 Script
- Running the RNHCEOG20 Script
- Running the RNCSUPG03 Script
- Retaining the Target Rename Log Files
- Running RNHCUPG01 Script on Copy of Current Demo
- Running RNHCEOG20 Script on Copy of Current Demo

- Running RNCSUPG03 Script on Copy of Current Demo

## Understanding Renaming Records and Fields

During development of new releases, PeopleSoft sometimes renames records, fields, or specific occurrences of a field on a record (recfield renames). In this task, you will execute scripts to rename those same objects in your Copy of Production and in the Copy of Current Demo.

With these commands, Data Mover renames the objects in the record and field definitions in Application Designer and then logs an entry on the table PSOBJCHNG. It also changes all references to these objects in pages and PeopleCode. This will not rename the objects on the database tables at this time.

Later in the upgrade, you will generate the SQL that will alter the tables on the database. This alter process reads PSOBJCHNG and will rename these tables and fields. The SQL generated to perform that task will be different depending on the build options you select and your database platform, however the result is the same. For record renames, the old table no longer exists and the new table contains the data from the old tables. For field and recfield renames, any affected tables will contain the new column with data from the old column; the old column no longer exists on the tables.

If a field rename does not go through this process, the alter SQL will not recognize it as a rename. After the alter, both old and new columns exist on the table and a data conversion process is required to copy the data from the old column to the new. This is an important distinction to make.

---

**Important!** It is very important to resolve any errors with these rename scripts. Do not skip any lines that error. It is not possible to recover from missed renames. The consequences of skipping a rename are evident later in the upgrade when you are in the middle of running data conversion programs.

---

A few different things could happen: the conversion program could error because the PeopleSoft system is expecting only the new column on the table, but you have both old and new, or you may lose data. Because of the rename, the PeopleSoft system expects the data to be handled in the SQL alter process. If the data doesn't move in the SQL alter process, and you don't write a data conversion program to move the data, the process drops the old column without having copied the data to the new column.

There are several advantages to using this rename process. Any references to the renamed records or fields in your customizations will also be modified. The number of differences on the compare reports is reduced. The SQL alter moves the data from old to new efficiently and no additional data conversion steps are required.

### Task 1-15-1: Running the RNHCUPG01 Script

This script will rename records, fields, and recfields on the Copy of Production database. These renames should only execute during the initial upgrade pass, not the move to production passes.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 1-15-2: Running the RNHCEOG20 Script

You only need to run this script if you had Directory Interface installed on your Copy of Production database.



This script will rename records, fields, and recfields on the Copy of Production database. These renames should only execute during the initial upgrade pass, not the move to production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Directory Interface	All	All

## Task 1-15-3: Running the RNCSUPG03 Script

This script will rename records, fields, and recfields on the Copy of Production database. These renames should only execute during the initial upgrade pass, not the move to production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-15-4: Retaining the Target Rename Log Files

In order to retain a copy of the log files for the preceding rename script steps run against the Copy of Production database, you must resave the logs for those steps with new file names. Otherwise, these logs will be overwritten by the following rename script steps run against the Copy of Current Demo database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-15-5: Running RNHCUPG01 Script on Copy of Current Demo

The RNHCUPG01.DMS script will rename records, fields, and recfields on the Copy of Current Demo database. These renames should only execute during the initial upgrade pass, not the Move to Production passes. Run this script in Data Mover user mode.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Copy of Current Demo	Initial	All	All	All

## Task 1-15-6: Running RNHCEOG20 Script on Copy of Current Demo

You only need to run this script if you had Directory Interface installed on your Copy of Production database.

The RNHCEOG20.DMS script will rename records, fields, and recfields on the Copy of Current Demo database. These renames should only execute during the initial upgrade pass, not the Move to Production passes. Run this script in Data Mover user mode.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Copy of Current Demo	Initial	Directory Interface	All	All

## Task 1-15-7: Running RNCSUPG03 Script on Copy of Current Demo

The RNCSUPG03.DMS script will rename records, fields, and recfields on the Copy of Current Demo database. These renames should only execute during the initial upgrade pass, not the Move to Production passes. Run this script in Data Mover user mode.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Copy of Current Demo	Initial	All	All	All

---

## Task 1-16: Comparing Customizations

This section discusses:

- Understanding Compare UPGCUST
- Running Compare UPGCUST
- Running UPGCUST Filter Script
- Reviewing UPGCUST Compare Log
- Restoring Copy of Current Demo

### Understanding Compare UPGCUST

In this task, you create a project on your Copy of Production database called UPGCUST. You execute a database compare of all comparable object types. This compare is run to identify all customizations on the Copy of Production database. The database compare occurs between your Copy of Production and the Copy of Current Demo database.

## Task 1-16-1: Running Compare UPGCUST

Before you start the compare, review the compare options and modify them based on your specific requirements.

If you decide not to modify the compare options, the objects are still compared. However, the results are only available online in Application designer and are not written into the compare reports. The compare reports are tools to help you review changed objects. Based on the report filters you select, you may need to review the action flags for other objects in the compare project in Application Designer.

For example, you can modify the compare options so the report contains customized objects that are present in your Copy of Production database but are absent from the Copy of Current Demo database. Alternatively, you can review these objects online, through Application Designer, after the database compare.

To run the database compare:

1. Using your current codeline, launch Application Designer and sign on to your Copy of Production database.
2. Select Tools, Upgrade, Compare and Report.
3. Enter UPGCUST as the project name in the Save Project As dialog box.
4. Click OK.

The PeopleSoft Sign on dialog appears.

5. Sign on to your Copy of Current Demo database.

The Compare and Report dialog box appears.

6. Click Options

The Upgrade Options dialog box appears.

7. Select the General Options tab.
8. Select the Set Target Audit Flags from Source radio button.
9. Select the Compare Options tab.
10. Select Database as the Compare Type.
11. Enter a directory path in the Report Output section.
12. Accept the defaults for all remaining compare options.
13. Select the Report Filter tab.

The default options include your custom changes on the reports.

14. Change the default options as necessary and click OK.
15. In the Upgrade Options dialog box, click OK.
16. Review the Object Type list.

All object types should be selected.

17. Click Compare to initiate the database compare.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-16-2: Running UPGCUST Filter Script

This step removes all objects from the UPGCUST project that are not marked *\*Changed* or *\*Unchanged* in your Copy of Production environment. It is used to isolate only custom objects in the UPGCUST project.

The script name for your upgrade is:

`PUUPX99.DMS`

See Appendix: “Using the Comparison Process.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-16-3: Reviewing UPGCUST Compare Log

In this step, review the log file and compare reports generated by the database compare in the previous step to ensure that it completed successfully. A detailed analysis of these compare reports is not necessary. Later in the upgrade, you will review a new set of compare reports when customizations are compared to the New Release Demo database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-16-4: Restoring Copy of Current Demo

Restore your Copy of Current Demo database from the backup taken earlier in the upgrade. The backup was taken before rename scripts ran against the Copy of Current Demo. This is done to restore the environment to an Oracle-delivered demo implementation.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Copy of Current Demo	Initial	All	All	All

## Task 1-17: Identifying Customizations

In this task, identify your modifications to Mass Change, EDI, Message Catalog, SQR Strings, XML Service Information, Verity Based Indexes, Setup Manager data, Pagelet Wizard objects, and related language system data, so that you can reload them later in the upgrade process.

The upgrade tasks will replace all Mass Change processes, Verity Based Indexes, and Setup Manager data. Only modifications to delivered Pagelet Wizard objects will be overwritten because any non-delivered custom Pagelet Wizard objects will be preserved during the upgrade. You cannot print Mass Change code. Be sure that you have extracted your modifications to reapply them later. You must extract your modifications, using cut and paste, to a file for manual reapplication later. EDI tables must be handled in the same way. Reload additional data and review customizations in Oracle-delivered data.

Message sets 0-19,999 will be overlaid during the upgrade, so any customizations you have made in this range will be lost. In addition, all SQR strings will be replaced. To save your customizations, cut and paste your changes to a file and manually reapply them.

Be aware that the data loaded by the PeopleSoft software must not be overwritten.

If you have multiple languages loaded, you should save any custom data you have in related language tables for system data. For these tables, data will be exported from the New Release Demo database when you export related language system data, and imported to your Copy of Production when you import related language system data. The import may delete your custom data, depending on the import option.

The tables that need to be reviewed are listed in the following scripts. These scripts can be found in your new release PS\_HOME\SCRIPTS directory.

**Important!** The scripts below are delivered with and run from your new PeopleSoft release. These scripts are *not* run in this task. You will run these scripts later in the upgrade process.

Review the tables that will be overwritten in these scripts:

Tables	Script
Message Catalog	DLUPX01E.DMS
SQR Strings	DLUPX04E.DMS
EDI	DLUPX05E.DMS
Mass Change	DLUPX06E.DMS
XML Service Information	DLUPX13E.DMS

Tables	Script
Setup Manager, Verity Based Indexes, and Optimization Models	DLUPX16E.DMS
Pagelet Wizard	DLUPX14E.DMS

If your database contains translations, review the list of related language system data tables that will be exported and imported in these scripts:

DLHCLASYSE.DMS

DLHCLASYSI.DMS

**Note.** Move to Production: Once you have reapplied these customizations at the end of your initial upgrade pass, you will not need to apply them again. The affected tables are moved from the old Copy of Production to the New Copy of Production by the scripts listed in the following table:

Tables	Scripts
Mass Change	MVAPPEXP.DMS MVAPPIMP.DMS
EDI	MVPRDEXP.DMS MVPRDIMP.DMS
Strings	MVAPPEXP.DMS MVAPPIMP.DMS
Messages	MVAPPEXP.DMS MVAPPIMP.DMS
XML Service Information	MVPRDEXP.DMS MVPRDIMP.DMS
Setup Manager, Verity Based Indexes, and Optimization Models	MVAPPEXP.DMS MVAPPIMP.DMS
Pagelet Wizard	MVUPX16E.DMS

## See Also

“Apply Application Changes,” Loading Data for Data Conversion.

“Apply Application Changes,” Loading Data to Complete System Setup.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 1-18: Preparing for the Application Upgrade

This section discusses:

- Running the XLATTABLE Copy Script
- Creating a Copy of RecField Definitions
- Dropping Student Financials Tables
- Copying Time and Labor Temp Table List
- Reversing Renames for Microsoft SQL Server
- Renaming Table with Data Type Changes for Microsoft
- Renaming DI Tables for Microsoft
- Renaming DI Tables for Oracle
- Renaming DI Tables for Informix
- Renaming DI Tables for Sybase
- Renaming DI Tables for DB2 zOS
- Renaming DI Tables for DB2 UNIX NT
- Editing the System Data Swap Script
- Editing the PeopleTools Swap Script

### Task 1-18-1: Running the XLATTABLE Copy Script

This script creates a temporary table PS\_UPG\_HR\_XLAT\_TMP containing translation values from the XLATTABLE for the field values that will be used during the data conversion.

The script name is:

PUHCHRG10.DMS

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 1-18-2: Creating a Copy of RecField Definitions

This step creates a copy of the contents of PSRECFIELD, before the upgrade is begun. It is used by the data conversion code to determine the structure of tables that may have been impacted by fixes you applied.

The script name is:

PUUPX07.DMS

---

**Note.** If you upgraded your system before, you may need to drop PSRECFIELD\_TMP prior to running this script.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 1-18-3: Dropping Student Financials Tables

The individual validation of the Service Indicator in the Student Financials disbursement process has been redesigned to meet the new standards for PeopleSoft 9.0. The Service Indicator has been replaced with Service Impacts. Therefore, the DISB\_RULE\_ITS and DISB\_RULE\_SRV tables have been modified to reflect the changes. Since the Service Indicator can be attached to multiple Service Impacts, the DISB\_RULE\_ITS and DISB\_RULE\_SRV tables will not be upgraded. You are required to add the appropriate service impact to the disbursement rules.

In this step, you drop the DISB\_RULE\_ITS and DISB\_RULE\_SRV tables, which as noted above will be recreated during this upgrade.

Run the following Data Mover script from the PS\_HOME\SCRIPTS directory:

```
PUCSSFR01.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	All	All

## Task 1-18-4: Copying Time and Labor Temp Table List

This script makes a copy of the temp tables list used in the process TL\_TIMEADMIN. After the upgrade is complete, you run a report to see which tables were removed from the new release. You then need to modify any of your custom Time and Labor Rules and Time Administration Application Engine steps that use the temp tables.

The script name is:

```
PUHCTLW01.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All



## Task 1-18-5: Reversing Renames for Microsoft SQL Server

During the PeopleTools upgrade, all tables will be altered for the new data type conversion. This alter will also alter the tables for the records and field renames run earlier in the upgrade. If a field has been deleted from a table in the new release, it would not normally be renamed, but this process will rename it. This will cause problems during data conversion when the original field name is being converted.

This step runs a script that will reverse those renames so that the PeopleTools data type conversion does not alter those tables for rename. The rename will be reapplied after the PeopleTools upgrade so that your compare to the New Release Demo database is consistent with your previous compare.

The script for your path is:

RNHCUPG04.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	MS SQL Server	All

## Task 1-18-6: Renaming Table with Data Type Changes for Microsoft

Run this step if you are using the MS SQL Server platform.

During the PeopleTools upgrade, tables will be altered for the new release. The table in this script will fail with an error because of an unsupported conversion, if it is not renamed. The data will be returned to the table in the proper data format during data conversion later in the upgrade.

The script for your path is:

PUHCUPG03MSS.SQL

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 1-18-7: Renaming DI Tables for Microsoft

Run this step if you have installed Directory Interface on MS SQL Server. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

PUHCEOG04MSS.SQL

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 1-18-8: Renaming DI Tables for Oracle

Run this step if you have installed Directory Interface on Oracle. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

```
PUHCEOG04ORA.SQL
```

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 1-18-9: Renaming DI Tables for Informix

Run this step if you have installed Directory Interface on Informix. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

```
PUHCEOG04IFX.SQL
```

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

## Task 1-18-10: Renaming DI Tables for Sybase

Run this step if you have installed Directory Interface on Sybase. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

PUHCEOG04SYB.SQL

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Sybase	All

## Task 1-18-11: Renaming DI Tables for DB2 z/OS

Run this step if you have installed Directory Interface on DB2 z/OS. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

PUHCEOG04DB2.SQL

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 1-18-12: Renaming DI Tables for DB2 UNIX NT

Run this step if you have installed Directory Interface on DB2 UNIX NT. There were PeopleTools tables added to the database in order for Directory Interface to work. These PeopleTools tables are now part of the core product. Renaming these tables now will preserve the data and prevent conflict during the PeopleTools upgrade.

The script for your path is:

PUHCEOG04DBX.SQL

If for some reason any of these tables or views do not exist in your database, and you receive SQL errors trying to rename them, this is OK and you can proceed with your upgrade.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2/UNIX NT	All

## Task 1-18-13: Editing the System Data Swap Script

This step should only be completed if your Copy of Production has a base language other than English. In a later step you will swap PeopleTools Managed Object tables that have related language on your New Release Demo database, so that it is translated correctly when you copy to your Copy of Production. In this step, you edit the swap script to set your New Release Demo database language to the same language as your Copy of Production.

To do this, follow the edit instructions in the script.

The swap system data script for your path is:

```
DLHCLASWAP.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	Non-English Base Language

## Task 1-18-14: Editing the PeopleTools Swap Script

This step should only be completed if your Copy of Production has a base language other than English. In a later step you will swap all PeopleTools Managed Object tables that have related languages on your Demo Database so that it is translated correctly when you copy to your Copy of Production. In this step you edit the swap script to set your Demo database language to the same language as your Copy of Production.

To do this, follow the edit instructions in the script.

The swap script for your upgrade path is:

```
PT_RELEASE_SWAP.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	Non-English Base Language

---

## Task 1-19: Backing Up After Prepare Your Database

Back up your Copy of Production database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remaining tasks in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All



## CHAPTER 2

# Apply PeopleTools Changes

This chapter discusses:

- Understanding PeopleTools Changes
- Verifying Upgrade User
- Performing Script Modifications
- Updating PeopleTools System Tables
- Turning Off Change Control
- Loading Model Definition Data
- Loading Message Data
- Renaming PeopleTools Records
- Comparing PeopleTools Objects
- Exporting and Copying Projects
- Populating Tablespace Data
- Building the PeopleTools Project
- Building Updated PeopleTools Project
- Migrating Records to New Tablespaces
- Loading Base Data
- Loading Language Data
- Loading PeopleTools Data
- Creating PeopleTools Views
- Converting PeopleTools Objects
- Converting Integration Broker
- Verifying NonBase Language Pages
- Clearing Process and Report Tables
- Setting Object Version Numbers
- Converting Database Data Types
- Backing Up After PeopleTools Upgrade

---

## Understanding PeopleTools Changes

In order to implement a successful upgrade, you must apply the necessary PeopleTools changes. This involves updating PeopleTools system tables, copying and building PeopleTools projects, loading PeopleTools seed data, and converting PeopleTools objects. From this point forward, you run all steps using your newly installed version of PeopleTools.

---

**Note.** Unless otherwise indicated, all scripts can be found in your new release PeopleSoft codeline PS\_HOME\SCRIPTS directory. The actual script name is indicated in the description of each step in uppercase letters.

---

---

### Task 2-1: Verifying Upgrade User

In this task, you verify that the user performing the upgrade steps has proper permissions to complete the upgrade.

Ensure that your upgrade user has PeopleSoft administrator privileges. This allows access to the PeopleSoft portal to make necessary security changes for the upgrade and to run the Portal Application Engine upgrade program. You use this ID to update the security setting for your other users so they can sign on after the upgrade.

---

**Warning!** You must perform this step now using your old version of PeopleTools. If you skip this step, or if your user has insufficient PeopleSoft administrator privileges, you will not be able to complete your upgrade. You cannot complete this step later in the upgrade process. Perform the following steps to grant administrator privileges now.

---

To grant your upgrade user PeopleSoft administrator privileges:

1. From the Go menu, select PeopleTools, Maintain Security.
2. Select Use, User Profiles, Administrator, Update/Display.
3. Select the user ID for your upgrade user.
4. Select the Is User System Administrator? check box on the user profile if it is cleared.
5. Save the user profile.
6. From Maintain Security, select Use, Permission Lists, General, Update/Display.
7. Select the primary permission list for your upgrade user.
8. Select the Can Start Application Server? check box if it is cleared.
9. Save the permission list.

See the Enterprise PeopleTools PeopleBook: Security Administration for your new release.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All



---

## Task 2-2: Performing Script Modifications

This section discusses:

- Understanding Script Modifications
- Copy the PTDDLUPG Script
- Edit the PTDDLUPG Script
- Run a DBTSFIX Report
- Edit the DBTSFIX Output Scripts
- Edit the GRANT Script
- Edit the RELGA Script
- Edit the PSLANG Script
- Edit the PSLANGUAGES Script
- Edit the TLSUPGNONCOMP Script
- Edit the PTxxxTLS Scripts
- Edit the DB2 Scripts
- Edit Move to Production Import Scripts
- Edit Move to Production Password
- Edit the DDLDB2 Script
- Edit the DDLDBX Script
- Edit the DDLORA Script
- Edit the DDLIFX Script
- Edit the MSGTLSUPG Script
- Edit the Integration Broker Script
- Edit Multilingual Step Properties
- Edit the Message Nodes Script
- Edit Data Type Steps

### Understanding Script Modifications

In this task, you perform preparation steps and make manual modifications to scripts delivered with your new PeopleSoft release. You must make the following modifications before proceeding with the remainder of your upgrade.

---

**Note.** Move to Production: These steps will be repeated in the Move to Production (MTP) pass. The script that you previously edited may be acceptable, or you may need to change it again if your New Copy of Production has a different security or data definition language (DDL) configuration.

---

## Task 2-2-1: Copy the PTDDLUPG Script

In this step, you copy the PTDDLUPG.SQL script to the PS\_HOME\SCRIPTS directory. If you are an Oracle/UNIX customer, transfer the file from the UNIX file server (PS\_HOME/SCRIPTS/UNIX) to your Windows file server PS\_HOME\SCRIPTS directory. If you are an Oracle/NT customer, you can find the file in PS\_HOME\SCRIPTS\NT.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle	All

## Task 2-2-2: Edit the PTDDLUPG Script

In this step, you edit PS\_HOME\SCRIPTS\PTDDLUPG.SQL to add site-specific tablespace names, database names, and STOGROUPs. PeopleTools delivers new tablespaces in the new PeopleSoft release to accommodate trees, application engine, temp tables, and tables that require row-level locking. Follow the instructions in the script for your platform. Since this script builds new tablespaces as part of the upgrade to the new PeopleSoft release, remove any tablespaces from the script that already exist in your database.

---

**Note.** DB2 UNIX/NT Unicode customers need to rename the PTDDLUPGU.SQL script to PTDDLUPG.SQL. If you choose not to rename the file, you will need to modify the step titled “Create Tablespaces” to run the PTDDLUPGU.SQL script. You can find this file in the PS\_HOME\SCRIPTS directory.

---



---

**Note.** DB2 z/OS customers need to edit the PTDDLUPG.SQL script generated during installation. This script needs to be placed in the PS\_HOME\SCRIPTS\ directory so it can be run later during the upgrade.

---

See Updating PeopleTools System Tables.

See Create Tablespaces.

---

**Note.** Informix customers need to edit the PS\_HOME\SCRIPTS\UNIX\PTDDLUPG.SH script.

---

See Create Tablespaces for Informix.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 2-2-3: Run a DBTSFIX Report

The DBTSFIX.SQR script aligns the tablespaces in the delivered release scripts with the Target database used during the upgrade. This process generates new release scripts, conforming to the RELxxxDBTSFIX.SQL naming convention that you run in a later task. Run this script to preserve your existing table-to-tablespace mapping in the Target database. The result of this task will be a RELxxxDBTSFIX.SQL script in which xxx represents a release number (for example, 800, 810, 811, 812, and so on) associated with your particular path.

---

**Note.** Before running this step, verify that the PS\_HOME values are set correctly in the Change Assistant environment for your upgrade job. Change Assistant uses the PS\_HOME information to determine which scripts need to be generated.

---



---

**Important!** Do not run the new release script at this point. You will be instructed to run this script later in the upgrade process.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 2-2-4: Edit the DBTSFIX Output Scripts

Edit the generated RELxxxDBTSFIX scripts according to the comments within each script. Verify that the data definition language (DDL) is accurate for your environment for tablespaces, database names, owner IDs, and so forth. The scripts can be found in your Change Assistant output directory for this upgrade path.

---

**Warning!** Do not run output scripts at this time. At this point in the upgrade process, you must only review the DBTSFIX output scripts.

---



---

**Note.** For DB2 z/OS customers only: When upgrading from one PeopleSoft release to the next, it is possible to move tables from a tablespace using a 4-KB buffer pool to one using a 32-KB buffer pool. The tablespaces PSIMAGE and PSIMGR use 32-KB buffer pools in Oracle-delivered applications. To maintain the tablespace schema used at your site, the DBTSFIX.SQR script will revise the upgrade scripts with the database and tablespace information from your database (the Target database). Tables assigned to tablespaces PSIMAGE or PSIMGR in the upgrade scripts are the exception to this approach. Note that Oracle has reassigned some tables to PSIMAGE or PSIMGR because they now require a 32-KB buffer pool. You must manually edit the “Create Table” statements in the upgrade scripts to replace the tablespace name PSIMAGE or PSIMGR with an appropriate tablespace name in your implementation that utilizes a 32-KB buffer pool. The database name must also be replaced with the value corresponding to the tablespace you are using.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 2-2-5: Edit the GRANT Script

Edit `PS_HOME\SCRIPTS\GRANT.SQL` and make the necessary modifications as documented in the script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-2-6: Edit the RELGA Script

Edit `PS_HOME\SCRIPTS\RELGA.DMS` and make the necessary OWNERID and synonym modifications as documented in the script.

The Update\_Oprdefn command in this script performs the following functions:

- Decrypts passwords in PSOPRDEFN, using old Data Encryption Standard (DES) algorithms, and re-encrypts using Secure Hashing Algorithm 1 (SHA-1) hashing.
- Decrypts passwords in PSACCESSPRFL using old keys and re-encrypts using new keys.
- Updates PSAUTHITEM to remove high-order bits from the Authorized Actions table.
- Updates permissions from Security Administrator to Maintain Security.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	DB2 z/OS	All

## Task 2-2-7: Edit the PSLANG Script

Edit `PS_HOME\SCRIPTS\PSLANG.DMS` and make the necessary modifications as documented in the script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-8: Edit the PSLANGUAGES Script

Edit `PS_HOME\SCRIPTS\PSLANGUAGES.DMS` and make the necessary modifications as documented in the script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-9: Edit the TLSUPGNONCOMP Script

Edit `PS_HOME\SCRIPTS\TLSUPGNONCOMP.DMS` and make the necessary modifications as documented in the script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-10: Edit the PTxxxTLS Scripts

This step applies only if you are running on a DB2 z/OS platform.

To edit the PTxxxTLS scripts:

1. Edit all of the scripts in the `PS_HOME\SCRIPTS` directory on the file server that conform to this file naming convention:

```
PTxxxTLS.DMS
PTxxxTLSyty.DMS
```

The `xxx` represents a PeopleTools release greater than your current PeopleTools release and `yyy` represents the three-letter language code.

2. Uncomment and modify the set owner ID command within each script, as in the following example:

```
set execute_sql set current sqlid = 'OwnerId In Upper Case';
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-11: Edit the DB2 Scripts

Perform this step only if your database platform is DB2 z/OS. DB2 z/OS scripts that create tables need the `set current sqlid` statement so that the tables are created with the correct owner ID. Open each script listed below, then uncomment and modify all of the DB2-specific statements to reflect your environment.

For SQL scripts, if the script does not contain DB2-specific statements, add the following line to the top of the script and edit it for your environment:

```
set current sqlid = 'OWNERID (in uppercase)';
```

For Data Mover scripts (DMSs), if the script does not contain DB2-specific statements, add the following line to the top of the script and edit it for your environment:

```
set execute_sql set current sqlid = 'OWNERID (in uppercase)';
```

Following is a list of the scripts that you need to edit:

```
DB2TMPIDXCREATE.SQL
DLUPX02I.DMS
DLUPX13I.DMS
DLUPX14I.DMS
DLUPX96I.DMS
PT_RELEASE_IMPORT.DMS
DB2ALLCCSIDUPD.SQL
DLHCSYSI.DMS
DLHCLASYSI.DMS
DLHCHRW01.DMS
DLHCCOG03I.DMS
DLHCUPG01.DMS
RNHCUPG03DB2.SQL
RNCSUPG02DB2.SQL
RNCSUPR04DB2.SQL
DLUPX16I.DMS
DLUPX95.DMS
DLHCUPT01I.DMS
DLHCUPT04I.DMS
DLCSFAG02I.DMS
DLHCUPW10I.DMS
```

---

**Note.** The DLUPX96I.DMS script will run on your Source database. Remember to edit this script for your Source database. All of the other scripts listed will run against the Target database.

---

In several steps in the upgrade process, project definitions are copied into the database. Any DB2 z/OS scripts that are built from these project definitions will need to be modified before you run them. Set the following steps in your Change Assistant job to a manual stop and edit the scripts for correct database/tablespace information. When you build the SQL scripts after copying the project, the database/tablespace names are the default values. You need to change these to the Target database specific values. To set a step as a manual stop in Change Assistant, highlight the step and select Edit, Stop from the menu bar.

In chapter 4, “Apply Application Changes,” set the step ReCreating Upgrade Tables (in the task Modifying the Database Structure) as a manual stop and edit the UPGCONVERT\_CRTTBL.SQL script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-12: Edit Move to Production Import Scripts

Perform this step only if your database platform is DB2 z/OS.

During the Move to Production, there are several scripts that export data from the previous Copy of Production to the New Copy of Production. These scripts export the tables to a DAT file. When the tables are exported, all the table attributes, including the database specific information (table owner, database name, and tablespace name), are stored in the DAT file. When you run the import script, it tries to create the tables and indexes using the database-specific information from the DAT file. So even though you ran the import script against your Copy of Production, you would still create tables in the upgraded database (which is the source database for the Move to Production step). To create the tables in the Target database, open each script listed below, then uncomment and modify all of the DB2-specific statements to reflect your environment.

You will also need to add the following command into MVPRDIMP.DMS, near the end of the script, just after the REPLACE\_DATA PSSTATUS command, but before the REPLACE\_VIEW PSTEMPBLCNTVW command, to change *ownerid* to the owner ID of your database.

```
Update PSSTATUS set OWNERID='OWNERID (in uppercase)';
```

Following is a list of the scripts that you need to edit:

```
MVAPPIMP.DMS
MVPRDIMP.DMS
MVHCPYW01I.DMS
MVHCPYT05I.DMS
MVHCPIT01I.DMS
MVHCHCW01I.DMS
MVHC80IMP.DMS
MVCS80IMP.DMS
```

If you prefer, you can copy these overrides from the xxDMODBO.DMS script that was generated from DBSetup while installing your database. Make sure you remove the SET NO RECORD if you copy from the DBSetup generated file.

See the Enterprise PeopleTools PeopleBook: Data Management for your new release.

See “Apply Changes to Production Database,” Performing the Move to Production.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	DB2 z/OS	All

## Task 2-2-13: Edit Move to Production Password

If your access ID and access password are different in the Copy of Production database than in the New Copy of Production database, you need to reset the access password in the MVPRDIMP.DMS script.

To modify passwords in your New Copy of Production database, append the following to your MVPRDIMP.DMS script and replace *ownerID*, *accessID*, and *accesspswd* with your values in the New Copy of Production database:

```
UPDATE PSSTATUS set OWNERID = 'ownerID';
UPDATE PSACCESSPRFL SET ACCESSID = 'accessID',
ACCESSPSWD = 'accesspswd', ENCRYPTED = 0;
ENCRYPT_PASSWORD *;
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 2-2-14: Edit the DDLDB2 Script

Edit PS\_HOME\SCRIPTS\DDLDB2.DMS. At the bottom of this script, you will see an insert into PSDDLDEFPARMS. This insert contains default information used when creating a table, an index, a unique index, or a tablespace. Verify with your database administrator that the last value for each row is appropriate for your environment by checking the values currently stored in your PSDDLDEFPARMS table. Otherwise, the values will be reset to the default values delivered by Oracle in this script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-15: Edit the DDLDBX Script

Edit PS\_HOME\SCRIPTS\DDLDBX.DMS. At the bottom of this script, you will see an insert into PSDDLDEFPARMS. This insert contains default information used when creating a table, an index, a unique index, or a tablespace. Verify with your database administrator that the last value for each row is appropriate for your environment by checking the values currently stored in your PSDDLDEFPARMS table. Otherwise, the values will be reset to the default values delivered by Oracle in this script.



### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

### Task 2-2-16: Edit the DDLORA Script

Edit `PS_HOME\SCRIPTS\DDLORA.DMS`. At the bottom of this script, you will see an insert into `PSDDLDEFPARMS`. This insert contains default information used when creating a table, an index, a unique index, or a tablespace. Verify with your database administrator that the last value for each row is appropriate for your environment by checking the values currently stored in your `PSDDLDEFPARMS` table. Otherwise, the values will be reset to the default values delivered by Oracle in this script.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

### Task 2-2-17: Edit the DDLIFX Script

Edit `PS_HOME\SCRIPTS\DDLIFX.DMS`. At the bottom of this script, you will see an insert into `PSDDLDEFPARMS`. This insert contains default information used when creating a table, an index, a unique index, or a tablespace. Verify with your database administrator that the last value for each row is appropriate for your environment by checking the values currently stored in your `PSDDLDEFPARMS` table. Otherwise, the values will be reset to the default values delivered by Oracle in this script.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

### Task 2-2-18: Edit the MSGTLSUPG Script

Edit `PS_HOME\SCRIPTS\MSGTLSUPG.DMS` and make the necessary modifications as documented in the script.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-2-19: Edit the Integration Broker Script

Edit `PS_HOME\SCRIPTS\PTIBUPGRADE.DMS` and make the necessary modifications as documented in the script. User level node security and transactional security have been added as of PeopleTools 8.48. Service namespace information, a low-level user on the node, and a low-level permission list for service operations, need to be specified. Consult with your Integration Broker specialist for assistance.

**Note.** If you are upgrading from PeopleTools 8.48 or later, this step and all of the steps in the task “Converting Integration Broker” do not need to be run because the Integration Broker conversion has already been performed. You may mark all of these steps as complete in your upgrade job. If you don’t mark these steps as complete, the upgrade will try to unnecessarily reconvert your objects.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-2-20: Edit Multilingual Step Properties

In this step, you will edit the Change Assistant step properties for the multilingual PeopleTools project copy step (or steps). Copy only the translated objects for the languages you license. This prevents the translated objects for unlicensed languages from copying over. You will copy any multilingual projects later in the upgrade process.

Depending on which languages you license, you will need to complete the following instructions once or twice. If you license any of these languages, Arabic, Czech, Danish, Finnish, French, Hebrew, Hungarian, Norwegian, Polish, Russian, or Turkish, perform the following instructions for the step “Export and Copy PPLTLSML Project.” If you license any of these languages, Canadian French, Dutch, German, Greek, Italian, Japanese, Korean, Malay, Portuguese, Simplified Chinese, Spanish, Swedish, Traditional Chinese, or Thai, perform the following instructions for the step “Export and Copy PPLTLS84CURML Project.”

To edit multilingual step properties:

1. In Change Assistant, select the step.
2. Open the Step Properties dialog box.
3. Click the Upgrade button, and then click the Options button.
4. On the Copy Options tab, deselect any languages that you do not license.  
Common and English should remain deselected.
5. Click OK three times.
6. Save the template in Change Assistant.

See Export and Copy PPLTLS84CURML Project.

See Export and Copy PPLTLSML Project.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All Non-English

## Task 2-2-21: Edit the Message Nodes Script

You need to run the PORTALNODES.DMS script before the PeopleTools upgrade UPG84PORTAL process.

In this step, you edit the PORTALNODES.DMS script. If you are using the Informix platform, edit the PORTALNODES.SQL script instead.

To edit the message nodes script:

1. Run the following SQL to determine your default local node value:

```
SELECT MSGNODENAME FROM PSMMSGNODEDEFN WHERE LOCALNODE = 1
```

2. Replace *all* instances of *PSFT\_PE* with the default local node name returned from the previous SQL.

---

**Note.** The values updated in this script must reflect the values for your Copy of Production database because that is the database being upgraded. This means that the database web server *hostname* and *pshome* values are the values that will exist for your upgraded web server for the upgraded database.

---

3. Replace *all* instances of *localhost* with the host name of the web server.

Include the port number and network domain name as needed. Unicode databases will need to add a *G* or *N* in front of the localhost name.

Example: `http://myserver/psc/pshome/`

Example: `http://myserver:8080/psc/pshome/`

Example: `http://myserver.mydomain.com:8080/psc/pshome/`

4. Replace *all* instances of *pshome* with the name of the directory where you installed the PeopleSoft web server files.

`http://myserver/psc/PA840DMO/`

---

**Note.** The 8.x content provider message nodes are: BP, CRM, EPM, ERP, HRMS, and SA.

---

5. Adjust the node type for the 8.x content provider message nodes.

In the 8.4 composite REL script, all content provider nodes have a node type of ICT to allow for the use of the 8.1x URL format. If a particular node name corresponds to a production application database that is being upgraded to PeopleTools 8.4, the message node type must be changed from ICT to PIA. (If the node type is not changed to PIA, the portal content references using that node will *not* be converted to the 8.4 simple URL format.)

If a particular node name corresponds to a production application database that will remain on PeopleTools 8.1x, then keep the node type as ICT. (If the node type is not left as ICT, the portal content references using that node will *not* be able to access the 8.1x application database.)

6. Replace *all* instances of *EMPLOYEE* with the default portal registry name if you need a different portal registry as the default; for example, *CUSTOMER* or *SUPPLIER*.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-2-22: Edit Data Type Steps

For PeopleTools 8.48 and later, new data types are supported for Microsoft SQL Server 2005 or later and Oracle. These data type changes are only available for use in conjunction with PeopleSoft application release 9.0 or later. If you have already converted data types or are upgrading to a PeopleSoft application release earlier than 9.0, you must mark these steps as complete in the template now. Do *not* run these steps unnecessarily.

To set the Data Conversion steps as complete:

1. In Change Assistant, select all the steps within the task Converting Database Data Types.
2. Press the F7 key.
3. Save the upgrade job in Change Assistant.

See Converting Database Data Types.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server Oracle	All

---

## Task 2-3: Updating PeopleTools System Tables

This section discusses:

- Understanding Updating PeopleTools System Tables
- Clean Up Message Data
- Create Tablespaces
- Create Tablespaces for Informix
- Update System Catalog Views
- Update PeopleTools System Tables
- Grant Privileges to the CONNECT ID
- Run the RELGA Script
- Populate Languages for Login
- Update the Product License Code

- Update PeopleTools Patch Information
- Create Temporary Performance Indexes
- Export PeopleTools System Tables
- Import PeopleTools System Tables
- Reset Database Options Flag
- Rerun Update Statistics for DB2 zOS
- Rerun RUNSTATS Report for DB2 UNIX NT
- Rerun Update Statistics for DB2 UNIX NT
- Rerun Update Statistics for Informix
- Rerun Update Statistics for Oracle

## Understanding Updating PeopleTools System Tables

In this task, you update your PeopleTools system tables by running various scripts.

---

**Important!** From this point forward, run all steps using the new release of PeopleTools on your Copy of Production database, unless otherwise indicated.

---

**Note.** If you license the PeopleSoft Rapid Apps Archiving Tool, read the appendix that describes it before beginning this step.

---

See Appendix: “Using the Data Archiving Tool.”

### Task 2-3-1: Clean Up Message Data

If you are upgrading from PeopleTools 8.47 or earlier, perform this step to clean out obsolete message data. Message functionality and structure changed as of PeopleTools 8.48 and the old data is obsolete. Edit PS\_HOME\SCRIPTS\PTUPGIBDEL.SQL to delete data from the tables that only exist in the old PeopleTools release. Open the script and make the following modifications:

1. Search for the string “--- End of PT8.1x/8.2x ---”.
2. Delete the entire portion of the script below this string.
3. Save the script as PS\_HOME\SCRIPTS\PTUPGIBDEL81X.SQL.

---

**Important!** Save the script using the naming convention shown above. This will preserve the original script for use in updating other databases at different PeopleTools releases, and assist in running the script automatically.

---

Follow this procedure to edit your template so that the script can run automatically:

1. Select this step and open the Step Properties dialog box.
2. Change the type from ManualStop to SQLScript, and click OK.
3. In your upgrade job, mark the step as Run.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-3-2: Create Tablespaces

This step runs the PTDDLUPG script, which builds new tablespaces as part of the upgrade to the new PeopleSoft release. PeopleTools delivers new tablespaces in this PeopleSoft release to accommodate trees, the application engine, temp tables, and tables that require row-level locking.

---

**Note.** DB2 UNIX/NT and DB2 z/OS Unicode customers: If you did not rename the PTDDLUPGU.SQL file when you edited the PTDDLUPG script, you must modify this step to run the PTDDLUPGU.SQL script. This file can be found in the \PS\_HOME\SCRIPTS directory.

---

See Edit the PTDDLUPG Script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle DB2 UNIX/NT DB2 z/OS	All

## Task 2-3-3: Create Tablespaces for Informix

FTP the PTDDLUPG.SH script file to the server. Log on as the database owner (Informix user) and run PTDDLUPG.SH to create the new tablespaces. This script creates new tablespaces introduced in the new PeopleSoft release. Tablespace changes have been made in order to group temp tables, take advantage of row-level locking, and group tables with similar attributes.

See Edit the PTDDLUPG Script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

## Task 2-3-4: Update System Catalog Views

This step runs the UPDOBJ.SQL script, which re-creates system catalog views that both DataMover and PeopleTools use.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Microsoft	All

## Task 2-3-5: Update PeopleTools System Tables

Release scripts are SQL scripts that modify the underlying table structure of a database so that it is compatible with a more recent PeopleTools release. They are located in the PS\_HOME\SCRIPTS directory. Release scripts can be identified by their common naming standard, RELxxx.SQL, in which xxx designates a PeopleTools release number.

These release (REL) scripts alter and update your PeopleTools tables to the current release. Change Assistant determines which RELxxx scripts to run based on the PeopleTools release of your Source and Target databases.

If you created RELxxxDBTSFIX (in which xxx is a PeopleTools release) earlier in your upgrade, the procedure will look at your Output folder and will know to run RELxxxDBTSFIX. If you did not run DBTSFIX, Change Assistant will run RELxxx.

---

**Note.** Before running this step, verify that the PS\_HOME values are set correctly in the Change Assistant environment for your upgrade job. Your new release PS\_HOME/SCRIPTS directory should contain all scripts that will be run during this step. This step runs at least one script. Do not proceed to the next step until these scripts run successfully.

---

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-3-6: Grant Privileges to the CONNECT ID

This step runs the GRANT.SQL script. This script grants select access to the connect ID for tables necessary for sign on.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-3-7: Run the RELGA Script

This step runs the RELGA.DMS script using bootstrap mode. All users will be captured in the log file. However, if a user ID has not been logged into the database before the upgrade, that user ID will be flagged in the log file. The message will state that the user's password cannot be decrypted and the password will be reset to the user's ID. This message occurs if the password column is null or blank. It will also occur if you have PSLOCK.SECURITY\_OPTION set to Y. In this case, after RELGA.DMS runs, all of your users' passwords (OPERPSWD) in PSOPRDEFN will be set to the user ID.

The Update\_Oprdefn command in Data Mover performs the following functions:

- Decrypts passwords in PSOPRDEFN using old DES algorithms and re-encrypts using SHA-1 hashing.
- Decrypts passwords in PSACCESSPRFL using old keys and re-encrypts using new keys.
- Updates PSAUTHITEM to remove high-order bits from the Authorized Actions table.
- Updates permissions from Security Administrator to Maintain Security.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-3-8: Populate Languages for Login

This step runs the PSLANG.DMS script, which populates the PSLANGUAGES table.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-3-9: Update the Product License Code

This section discusses:

- Understanding the Product License Code
- Updating the Product License Code

### Understanding the Product License Code

The new PeopleSoft release stores your application product license code on the database. This code is used to unlock the pages and Application Engine programs that you licensed. It also provides necessary product information about your database to be used for identifying software maintenance that may need to be applied.

You need to populate the databases that were upgraded to the new PeopleSoft release so that you have the correct access to pages and Application Engine programs that you licensed.

When your new PeopleSoft databases were installed, the appropriate application license code was added to your database in the PSOPTIONS table. This was done in an update statement that was created when DBSETUP was run to create the Data Mover script for the new PeopleSoft release. The location of this script is:



PS\_HOME\SCRIPTS\DBnameDBplatform.DMS

*DBname* is the name of the Demo database you installed and *DBplatform* represents the database platform using the following chart:

Database Platform	Code Used
Microsoft SQL Server	MSS
DB2 UDB z/OS	DB2
DB2 UDB UNIX/NT	DBX
Oracle	ORA
Informix	INF
Sybase	SYB

## Updating the Product License Code

Follow the steps below to update your product license code.

To update the product license code:

1. From the Data Mover script that was created for your new PeopleSoft database installation, copy out the update to PSOPTIONS.

The statement should look similar to this:

```
update PSOPTIONS set LICENSE_CODE = 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx', LICENSE_⇒
GROUP = 'xx' ;
```

where 'xx' equals your license code and 'xx' equals your license group.

2. Run the SQL command identified above using your SQL tool.

Your database is now updated with the appropriate license code. You can now access the pages and Application Engine programs that you licensed.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-3-10: Update PeopleTools Patch Information

In this step, you update your database with the version of the PeopleTools patch being applied.

---

**Note.** You only need to run this step if you are applying a PeopleTools patch as part of the upgrade process.

---

Log on to Data Mover in user mode and run the %PS\_HOME\SCRIPTS\PTPATCH.DMS script.

Review the PeopleTools patch instructions and perform any additional database upgrade instructions that may be listed prior to the copy of the patch project. The patch project will be copied later during the upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-3-11: Create Temporary Performance Indexes

Perform this step only if you are running on a DB2 z/OS platform. This step runs the DB2TMPIDXCREATE script to create multiple indexes for rename performance. You will drop these indexes later in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	DB2 z/OS	All

## Task 2-3-12: Export PeopleTools System Tables

The script for this step exports the content of the PeopleTools tables from the Copy of Production database during your Move to Production passes. During the initial pass, you run programs to convert some objects, like PeopleCode and fields. You perform analysis to decide which objects, such as records and menus, to bring over to your production database and which customized objects to keep. At the end of the initial pass, you reapply customizations or make other changes, such as modifying your permission lists. You do not need to repeat those tasks in the Move to Production pass because this script exports all of your changes to the PeopleTools objects.

The script name for your upgrade path is:

MVPRDEXP.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	All	All	All

## Task 2-3-13: Import PeopleTools System Tables

The script for this step imports the content of the PeopleTools tables into your New Copy of Production database during your Move to Production passes.

These MVPRD\* scripts replace tasks and steps performed in the initial pass. These tasks and steps may include:

- Converting PeopleTools Objects
- Renaming Records and Fields
- Running New Release Compare Reports

- Running New Release Upgrade Copy

If your relational database management system (RDBMS) uses tablespaces, edit this script for the proper DDL information.

The script name for your upgrade path is:

MVPRDIMP.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 2-3-14: Reset Database Options Flag

This step runs UPGDBOPTIONS\_DISABLE.SQL, which resets the PSSTATUS.UPGDBOPTIONS flag. The flag is reset only for upgrades where you are coming from a PeopleSoft application release prior to 9.0 and going to a PeopleSoft application release of 9.0 or later with PeopleTools 8.48 or later. The PeopleTools upgrade must be applied using the old data types as the data type conversion will occur after the PeopleTools changes have been completed.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	MS SQL Server Oracle	All

## Task 2-3-15: Rerun Update Statistics for DB2 zOS

Earlier in the upgrade process, you updated your statistics for DB2 z/OS. Due to changes in the database structure, you must update statistics again to improve the performance of your compare and copy. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-3-16: Rerun RUNSTATS Report for DB2 UNIX NT

This script creates the RUNSTATS.DAT file for the script to update the statistics for DB2 UDB on UNIX, Linux, or Windows.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

## Task 2-3-17: Rerun Update Statistics for DB2 UNIX NT

Earlier in the upgrade process, you updated your statistics for DB2 UDB on UNIX, Linux, or Windows. Due to changes in the database structure, you must update statistics again to improve the performance of your compare and copy. This step runs `RUNSTATS .SQL` to update statistics on your database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

## Task 2-3-18: Rerun Update Statistics for Informix

Earlier in the upgrade process, you updated your statistics for Informix. Due to changes in the database structure, you must update statistics again to improve the performance of your compare and copy. This step runs `UPDATESTATS` to update statistics on your database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

## Task 2-3-19: Rerun Update Statistics for Oracle

Earlier in the upgrade process, you updated your statistics for Oracle. Due to changes in the database structure, you must update statistics again to improve the performance of your compare and copy. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

---

## Task 2-4: Turning Off Change Control

This task executes a SQL statement that turns off the Change Control feature to improve performance for the upgrade copy. One of the tasks for completing database changes will remind you to turn this feature on again, if you want to use it.

---

**Note.** Move to Production: The Change Control feature slows down copy functions. The large copy projects are executed only during the initial pass and the feature is disabled only for the initial pass.

---

See “Complete Database Changes,” Reviewing Change Control.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 2-5: Loading Model Definition Data

This section discusses:

- Understanding Model Definition Data Load Process
- Load Model Definitions for DB2 zOS
- Load Model Definitions for DB2 UNIX NT
- Load Model Definitions for Oracle
- Load Model Definitions for Informix
- Load Model Definitions for Microsoft
- Load Model Definitions for Sybase

### Understanding Model Definition Data Load Process

In this task, you load model definition scripts for your database platform and populate DDL model definitions. This step runs the DDL model definition script applicable to your database platform. If required by your database platform, you modified this script in the task “Performing Script Modifications,” to use your site-specific information.

See Performing Script Modifications.

### Task 2-5-1: Load Model Definitions for DB2 zOS

This step runs the DDLDB2.DMS script to populate DDL model definitions for the DB2 z/OS platform.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-5-2: Load Model Definitions for DB2 UNIX NT

This step runs the DDLDBX.DMS script to populate DDL model definitions for DB2 UDB on UNIX, Linux, or Windows.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

## Task 2-5-3: Load Model Definitions for Oracle

This step runs the DDLORA.DMS script to populate DDL model definitions for the Oracle platform.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-5-4: Load Model Definitions for Informix

This step runs the DDLIFX.DMS script to populate DDL model definitions for the Informix platform.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

## Task 2-5-5: Load Model Definitions for Microsoft

This step runs the DDLMSS.DMS script to populate DDL model definitions for the Microsoft SQL Server.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-5-6: Load Model Definitions for Sybase

This step runs the DDLSYB.DMS script to populate DDL model definitions for the Sybase platform.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Sybase	All

---

## Task 2-6: Loading Message Data

This step loads system messages in the message catalog.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 2-7: Renaming PeopleTools Records

This process renames language-sensitive PeopleTools tables to coordinate with PeopleSoft standards for related language models. If you reference these tables in your customized PeopleCode, COBOL, SQR, Crystal, or queries, you need to evaluate and possibly change these references to the new record names. The name of the script for your upgrade path is:

`TLSRENAME.DMS`

The PeopleTools tables will be renamed as follows:

Previous Release	Current Release
XLATTABLE	PSXLATITEM
XLATTABLEDEL	PSXLATDEFNDEL

Previous Release	Current Release
PS_MESSAGE_SET_TBL	PSMSGSETDEFN
PS_MESSAGE_CATALOG	PSMSGCATDEFN

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-8: Comparing PeopleTools Objects

This section discusses:

- Understanding Comparing PeopleTools Objects
- Compare Fields in the PPLTOOLS Project
- Review the Field Compare Report

## Understanding Comparing PeopleTools Objects

In this task, you compare PeopleTools fields and identify any fields you may have changed. Then you review your field compare report.

See Appendix: “Understanding the Comparison Process.”

**Note.** If you are running Sybase, check the configuration parameter for open objects. If this parameter is set too low, you may encounter the following error: `ct_connect(): network packet layer: internal net library error` during the compare or copy process. If you encounter this error, you need to increase the value accordingly.

## Task 2-8-1: Compare Fields in the PPLTOOLS Project

This step applies only if you have modified PeopleTools fields. However, because fields are sometimes inadvertently modified, PeopleSoft recommends that all upgrade customers perform this step to ensure that all PeopleTools field modifications are identified.

The fields delivered by PeopleTools should not be modified in any way. Doing so may result in system instability or crashes in your PeopleSoft application. However, it is possible that a PeopleTools field could become modified from its delivered state. Compare the fields in the PPLTOOLS and PPLTLS84CUR projects to identify any fields that were inadvertently modified. It is also possible that PeopleTools delivered a new field that conflicts with a field you added to your PeopleSoft system. If you modified any fields delivered in the PeopleTools project, you need to identify those fields using the steps below so that any data in the modified field is converted appropriately to match the PeopleTools-delivered field definition or format. This is especially necessary if the size of the field changed. You *must* take the PeopleSoft-delivered field definition or your system may become unstable.



The PPLTOOLS and PPLTLS84CUR projects that you will copy in the next task contain changes to all the PeopleTools objects. You must copy these projects in their entirety for the system to work properly. If you used any of the fields in the projects in any of your modifications, these field changes may negatively impact your data. To ensure that this won't happen, you should compare the fields to your Copy of Production.

---

**Note.** In order for PeopleTools to operate correctly, any customizations to PeopleTools fields must not be retained during the copy process.

---

To compare fields to your copy of production:

1. Sign on to the Demo database using the new PeopleTools.
2. From the Go menu, select Application Designer.
3. Select File, Open....
4. From the Definition drop-down list box, select Project and click Open to display a list of all projects.
5. Select the PPLTOOLS project and click Open again.
6. Select File, Save Project As...
7. Enter *COMPARE\_PPLTOOLS* and click OK.
8. If the PPLTLS84CUR project does not exist on your Demo (source) database, then complete this step to copy the project definition; otherwise, continue to the next step.
  - a. Select Tools, Copy Project, From File...
  - b. In the resulting dialog box, change the import directory to *PS\_HOME\PROJECTS*.
  - c. Select *PPLTLS84CUR* from the list of projects and click Open.
  - d. In the Copy dialog box, click the Deselect All button.  
Make sure that no object types are selected, since only the project definition is loaded at this time.
  - e. Click the Copy button.  
When the progress dialog disappears, the project definition has been copied.
  - f. Select File, Open....
  - g. From the Definition drop-down list box, select Project and click Open to display a list of all projects.
  - h. Select the *COMPARE\_PPLTOOLS* project and click Open again.
9. Select File, Merge Projects... and click Insert to display a list of all projects.
10. Select the PPLTLS84CUR project and click Insert.  
This merges the contents of the PPLTLS84CUR project and the *COMPARE\_PPLTOOLS* project.
11. Click Close.
12. Select Tools, Compare and Report...
13. Sign on to your Copy of Production database.
14. From the Definition drop-down list box, select *Fields*.  
This is the only necessary comparison.
15. Click Options.
16. Verify that the Compare Report Output Directory is set to a valid location.
17. Click OK.

18. Select Compare.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 2-8-2: Review the Field Compare Report

Review your Field Compare Report. If you changed any PeopleTools fields, they will show up in the compare report as custom changed differences.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

---

## Task 2-9: Exporting and Copying Projects

This section discusses:

- Understanding Exporting and Copying Projects
- Export and Copy PTPORTALDEL Project
- Export and Copy PPLTOOLS Project
- Export and Copy PPLDELETE Project
- Export and Copy PPLTLS84CUR Project
- Export and Copy PPLTLS84CURML Project
- Export and Copy PPLTLSML Project
- Export and Copy PPLTLS84CURDEL Project
- Export and Copy PATCH84X Project
- Export and Copy PATCH84XML Project

## Understanding Exporting and Copying Projects

In this task, you export and copy projects.

PeopleSoft recommends that you verify the results of all copied projects. After a project has been copied, each object is identified with a checkmark in the Done column. You can view these results from the Upgrade tab in Application Designer. It is also recommended that you copy the PeopleTools projects with the take action flags set as they originally were set when the database was delivered.

---

**Note.** If you are running Sybase, check the configuration parameter for “open objects.” If this parameter is set too low, you may encounter the following error: `ct_connect(): network packet layer: internal net library error` during the compare or copy process. If you encounter this error, you will need to increase your parameter accordingly.

---

See the Enterprise PeopleTools PeopleBook: PeopleSoft Application Designer for your new release.

## Task 2-9-1: Export and Copy PTPORTALDEL Project

This step deletes obsolete or changed Portal registry structures. These obsolete or changed structures must be deleted before the new Portal registry structures are copied, when you export and copy the PPLTOOLS project.

See Export and Copy PPLTOOLS Project.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-2: Export and Copy PPLTOOLS Project

This process copies specified tools objects to your database that are necessary for the proper operation of PeopleTools.

Before the fields are copied, the upgrade process detects whether the object definition exists or not. PeopleSoft delivers the PPLTOOLS project with an action of `CopyProp` to prevent the possible overwrites of custom field labels. When the upgrade process detects that a given field does not exist, it changes that action so that the entire definition can be copied. You can ignore any errors you receive at this time similar to the following examples:

```
Changed Action from CopyProp to Copy, definition does not exist on target.
Definition Name: OBJECTNAME not copied, entire definition already copied.
```

These warnings occur because the PeopleTools project contains fields along with their field label. This is necessary so the software does not overwrite any customized field labels on PeopleSoft field objects.

PeopleSoft deletes objects that existed in earlier PeopleTools releases. Therefore, you will see warnings when copying the PPLTOOLS project from your source database that you upgraded to the new PeopleSoft release. This project may contain objects that are orphaned because their parent objects were removed. You can ignore these warnings and continue with your upgrade. For example:

```
Definition Name: object_name not copied, not found on source database.
Definition Name: object_name not copied, parent definition does not exist in⇒
target database.
```

You may see the following error during the copy of Portal Registry Structures:

```
Duplicate Key. Portal: portalname, Obj name: objectname, CP: nodename, URL (1st 50⇒
char): URL
```

This error occurs when copying a Portal Registry Structure that has a different PORTAL\_OBJNAME but the same PORTAL\_URLTEXT as an existing registry object. If you get this error, you should take the PeopleSoft-delivered version of this object, but you need to evaluate whether you want to keep the conflicting object. Delete or modify the old object to remove the conflict and recopy this object to bring in the PeopleSoft-delivered version of this object.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-3: Export and Copy PPLDELETE Project

This process deletes PeopleTools objects that are no longer required for the new PeopleTools release. The fields in the PPLDELETE project are considered PeopleTools fields, but they are sometimes used in application tables. You will receive some errors at this time, similar to the following example:

```
Error: Field Fieldname is in use in at least one record.
```

These errors may be ignored because they may refer to fields that PeopleTools no longer uses in the new release but are fields that are still in use by PeopleSoft Applications. If these fields are not used in customizations, they will be deleted during the application compare and copy process. The following fields may be affected:

- RUN\_CNTL\_WRK
- ADDL\_TAXES
- DELETE\_RECORD
- EMPLOYER\_APPROVED
- YEAR\_START\_DT

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-4: Export and Copy PPLTLS84CUR Project

This process copies specified objects to the database that are necessary for the proper operation of PeopleTools. The PPLTLS84CUR project contains all PeopleTools objects that have been created or updated since PeopleTools 8.40 was released.

Before the copy of records and fields, the upgrade process detects if the object definition exists or not. PeopleSoft delivers the PPLTLS84CUR project with an action of CopyProp to prevent the possible overwrites of custom field labels and recfields. When the upgrade process detects that a given field or record does not exist, it changes that action so that the entire definition can be copied. You can ignore any errors that you may receive at this time similar to the following examples:

```
Changed Action from CopyProp to Copy, definition does not exist on target.
Definition Name: OBJECTNAME not copied, entire definition already copied.
```

These warnings occur because the PeopleTools project contains fields along with their field label. This is necessary so that the software does not overwrite any customized field labels on PeopleSoft field objects.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-5: Export and Copy PPLTLS84CURML Project

This process copies language-specific PeopleTools objects to the database that are necessary for the proper operation of PeopleTools.

Before the copy of records and fields, the upgrade process detects if the object definition exists or not. PeopleSoft delivers the PPLTLS84CURML project with an action of `CopyProp` to prevent the possible overwrites of custom field labels. When the upgrade process detects that a given field does not exist, it changes that action so that the entire definition can be copied. You can ignore any errors that you may receive at this time similar to the following example:

```
Changed Action from CopyProp to Copy, definition does not exist on target.
Definition Name: OBJECTNAME not copied, entire definition already copied.
```

This warning occurs because the PeopleTools project contains fields along with their field label. This is necessary so that the software does not overwrite any customized field labels on PeopleSoft field objects.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All Non-English

## Task 2-9-6: Export and Copy PPLTLSML Project

This process copies language-specific PeopleTools objects to the database that are necessary for the proper operation of PeopleTools.

Before copying records and fields, the upgrade process detects whether the object definition exists. PeopleSoft delivers the PPLTLSML project with an action of `CopyProp` to prevent the possible overwrites of custom field labels and recfields. When the upgrade process detects that a given field or record does not exist, it changes that action so that the entire definition can be copied. You can ignore any errors that you may receive at this time similar to the following examples:

```
Changed Action from CopyProp to Copy, definition does not exist on target.
Definition Name: OBJECTNAME not copied, entire definition already copied.
```

These warnings occur because the PeopleTools project contains fields along with their field labels. This is necessary so that the PeopleSoft system does not overwrite any customized field labels on PeopleSoft field objects.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	Arabic Czech Danish Finnish French Hebrew Hungarian Norwegian Polish Russian Turkish

## Task 2-9-7: Export and Copy PPLTLS84CURDEL Project

This process deletes specified PeopleTools objects from your database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-8: Export and Copy PATCH84X Project

This process copies specified objects to the database that are necessary for the proper operation of PeopleTools.

---

**Note.** Perform this process only if you are applying a PeopleTools patch that includes a database project. Check the patch documentation to verify whether a database project was delivered with the patch.

---

Use Application Designer to perform the following steps.

To export and copy the project:

1. Log on to your Target database as a valid user.
2. Select Tools, Copy Project, From File...
3. In the Import Directory field, enter *PS\_HOME\PROJECTS*.
4. Select the PATCH84X project from the list, in which 84X corresponds to the PeopleTools release of the patch project.

This should correspond with the PeopleTools release to which you are upgrading.

5. Select Options; select the languages Common and English *only*.
6. Select Copy to begin copying updated PeopleTools objects to your database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-9-9: Export and Copy PATCH84XML Project

This process copies language-specific PeopleTools objects to your database that are necessary for the proper operation of PeopleTools.

---

**Note.** Perform this process only if you are applying a PeopleTools patch that includes a database project. Check the patch documentation to verify whether a multilingual database project was delivered with the patch.

---

Use Application Designer to perform the following steps.

To export and copy the project:

1. Log on to your Target database as a valid user.
2. Select Tools, Copy Project, From File...
3. In the Import Directory field, enter *PS\_HOME\PROJECTS*.
4. Select the PATCH84XML project from the list, where 84X represents the PeopleTools release of the patch project.

This should correspond to the PeopleTools release to which you are upgrading.

5. Click Options, and then select the Copy Options tab and verify that only the non-English languages that are installed are selected.

---

**Note.** The languages English and Common should not be selected.

---

6. Click OK.
7. Click Copy.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All Non-English

## Task 2-10: Populating Tablespace Data

This section discusses:

- Create Application Tablespaces
- Populate Tablespace Data
- Update Tablespace Names

## Task 2-10-1: Create Application Tablespaces

During each Move to Production pass, you must create any new tablespaces. You can reuse the same script created during the initial pass when you created new tablespaces, or you can build a new one if you plan to use different tablespaces on your production system.

See “Apply Application Changes,” Updating Database Overrides, Create New Tablespaces.

The script supplied by Oracle to create tablespaces for your upgrade is:

```
HCDDL.SQL (HCDDL.U.SQL for DB2 z/OS Unicode)
```

---

**Important!** For DB2 UNIX/NT sites, the script name is HCDDL.DMS.SQL for ANSI, and HCDDL.DMSU.SQL for Unicode.

---

Once you have determined which script to run during Move to Production, you can change this step to run automatically.

To run the Create Application Tablespaces step automatically:

1. In Change Assistant, open your upgrade job.
2. In the task Populating Tablespace Data, right-click on the step Create Application Tablespaces, and then select Step Properties.
3. In the Step Properties dialog box, change the value in the Type field from *ManualStop* to *SQLScript*.
4. In the Script/Procedure field, enter the name of the script you want to run and click OK.
5. Select Edit, Run.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 2-10-2: Populate Tablespace Data

This step populates all tablespace information in the PSRECTBLSPC table. This step runs the SETSPACE.SQR script, which ensures that the correct tablespace information is populated for tasks later in the upgrade process.

The values stored in the DDLSPACENAME field are updated with current values found in the system catalog for tables already defined in your database. If you modified tablespace names from the Oracle-delivered names, this step makes those same changes in the PeopleSoft record definition.



If you receive any errors when you run this script, correct them by creating the needed tablespace or changing the tablespace definition on the record object. Then run the script again to validate that you have created all tablespaces.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 2-10-3: Update Tablespace Names

The SETSPACE SQR script identifies the tables with an invalid database name/tablespace combination. However, the PeopleTools Metadata tables in your Copy of Production (Target) database contain the database/tablespace values from the Demo (Source) database. This also occurs if your Demo and Copy of Production databases are in the same DB2 subsystem after the upgrade/copy is completed. SETSPACE.SQR corrects these values for those tables defined in DB2. For those tables that are defined in the PeopleTools metadata tables, but have not been defined in DB2, you need to review the SETSPACE SQR script for those tables that are reported as not defined in the database, but where the database/tablespace combination is valid. If the report shows an invalid database/tablespace combination, or shows your Demo (Source) database and tablespace names instead of your Copy of Production (Target) database and tablespace names, you can correct the database and tablespace names using one of the following options:

- Generate the alter/create scripts and globally edit the scripts, changing the database/tablespace values to those of your Copy of Production.
- Directly update the PSRECTBLSPC table with your Target database names before generating the alter/create scripts.

This will ensure that the database name/tablespace names in the generated alter/create scripts will be correct. The syntax to update the PSRECTBLSPC table is as follows:

```
UPDATE PSRECTBLSPC SET DBNAME = dbname, DDLSPACENAME = tablespace name WHERE⇒
DDLSPACENAME = tablespace identified in SETSPACE OUTPUT AND DBNAME = database⇒
identified in SETSPACE OUTPUT;
```

If you are using the Oracle-delivered tablespaces, you can omit the references to DDLSPACENAME in the SQL statement above.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 2-11: Building the PeopleTools Project

This section discusses:

- Generate the PeopleTools Script
- Edit the PeopleTools Script
- Run the PeopleTools Script

### Task 2-11-1: Generate the PeopleTools Script

This step generates the PeopleTools Create and Alter script to create and alter records of the type Table that are delivered in the PPLTOOLS project. The tables are altered to add new columns, rename existing columns, and change columns that have modified properties, such as length, and delete columns. The script also creates new indexes, re-creates modified indexes, and creates triggers. The script name is:

```
PTCREATEANDALTERNETABLES.SQL
```

---

**Note.** You may receive the following error: Error: WEBLIB\_XMLTOCI - Fatal error trying to open this record. Record not processed. (76,45).

This error occurs for any records deleted in the PPLTLS84CURDEL project. Ignore this error and continue with the upgrade.

---



---

**Note.** For DB2 z/OS sites, if this step takes an exceptionally long time, performing a RUNSTATS on the System Catalog tablespace SYSDATABASE may improve performance.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 2-11-2: Edit the PeopleTools Script

In this step, you edit the script you generated in the previous task for tablespace names and sizing. If you are running on an RDBMS platform that uses tablespaces, and you are *not* using the PeopleSoft tablespace names, have your database administrator review these scripts and modify the tablespace names appropriately.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

## Task 2-11-3: Run the PeopleTools Script

In this step, you run the script you generated in the previous step to create all records of the type Table. This step creates new table structures, alters existing PeopleSoft table structures, creates new indexes, re-creates modified indexes, and creates triggers.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-12: Building Updated PeopleTools Project

This section discusses:

- Generate Updated PeopleTools Script
- Edit the Updated PeopleTools Script
- Run the Updated PeopleTools Script

## Task 2-12-1: Generate Updated PeopleTools Script

This step generates the SQL script to create and alter records of the type table that are delivered in the PPLTLS84CUR project. The tables are altered to add new columns, rename existing columns, and change columns that have modified properties, such as length, and delete columns. The script will also create new indexes, re-create modified indexes, and create triggers. The script name is:

```
PPLTLS84CURTABLES.SQL
```

---

**Note.** For DB2 z/OS sites, if this step takes an exceptionally long time, performing a RUNSTATS on the system catalog tablespace SYSDBASE may improve performance.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-12-2: Edit the Updated PeopleTools Script

In this step, you edit the PPLTLS84CURTABLES.SQL script that was generated in the previous step for tablespace names and sizing. If you are running on a relational database management system (RDBMS) platform that uses tablespaces, and you are *not* using the PeopleSoft tablespace names, have your database administrator review this script and modify the tablespace names appropriately. The script can be found in your Change Assistant output directory for this upgrade path.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

## Task 2-12-3: Run the Updated PeopleTools Script

This step runs the script you generated in this task to create all records of the type table. This creates new table structures, alters existing PeopleSoft table structures, creates new indexes, re-creates modified indexes, and creates triggers.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-13: Migrating Records to New Tablespaces

This section discusses:

- Understanding Migration of Records to New Tablespaces
- Export and Copy the PT84TBLSPEC Project
- Build the Tablespace Alter Script
- Edit the Tablespace Alter Script

- Run the Tablespace Alter Script

## Understanding Migration of Records to New Tablespaces

In this task you migrate the tables delivered in the PT84TBLSPC project to the correct tablespaces.

### Task 2-13-1: Export and Copy the PT84TBLSPC Project

This process copies the records that moved to different tablespaces in the new release of PeopleTools. The upgrade copy options are set to Copy From Source for record DDL to pick up the new tablespace information.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

### Task 2-13-2: Build the Tablespace Alter Script

This step generates the SQL script to alter records of the type table that are delivered in the PT84TBLSPC project. The tables are altered to move them to the correct tablespaces for the new release of PeopleTools. The script name is:

TABLESPACEALERTABLES.SQL

---

**Note.** For DB2 z/OS sites, if this step takes an exceptionally long time, performing a RUNSTATS on the system catalog tablespace SYSDBASE may improve performance.

---

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

### Task 2-13-3: Edit the Tablespace Alter Script

In this step, you edit the TABLESPACEALERTABLES.SQL script for tablespace names and sizing. If you are running on an RDBMS platform that uses tablespaces, and you are *not* using the PeopleSoft tablespace names, you need to review and modify the scripts above. Have your database administrator review these scripts and modify the tablespace names appropriately. The script can be found in your Change Assistant output directory for this upgrade path.

---

**Note.** If you are a DB2 z/OS customer, you must edit the scripts for database name regardless of whether you are using the delivered PeopleSoft tablespace names.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

## Task 2-13-4: Run the Tablespace Alter Script

This step runs the TABLESPACEALERTABLES.SQL script to move the tables to the new tablespaces.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

---

## Task 2-14: Loading Base Data

These Data Mover scripts (DMSs) initialize and modify the data in various PeopleTools tables required for the system to execute properly. This step runs scripts conforming to the PTxxxTLS.DMS and PTxxxTLYyy.DMS naming conventions, where xxx represents a PeopleTools release number and yyy represents a three-letter language code, that are greater than your current PeopleTools release. For some upgrades, no data scripts are required. In this case, Change Assistant continues to the next step without producing a log file.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-15: Loading Language Data

This section discusses:

- Populate Languages
- Load Language Data

### Task 2-15-1: Populate Languages

This step runs the PSLANGUAGES.DMS script. This script populates the PSLANGUAGES table with Verity Locale data and other language-specific data.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 2-15-2: Load Language Data

If your database has languages installed in addition to English, you must populate the PSLANGUAGES table.

To load language data:

1. From the DMS that was created for your PeopleSoft 8.x database installation, find the UPDATE to PSLANGUAGES.

The statement should look similar to the following:

```
UPDATE PSLANGUAGES SET INSTALLED=1 WHERE LANGUAGE_CD = 'xxx' ;
```

2. Run the SQL command identified above using your SQL tool.

Your database is now updated with the language data.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All Non-English

## Task 2-16: Loading PeopleTools Data

This section discusses:

- Load NonComparable Objects
- Load English Messages
- Load English String Data

- Load Stored Statements Data

## Task 2-16-1: Load NonComparable Objects

This step runs the TLSUPGNONCOMP.DMS script. This script loads the TLSUPGNONCOMP project and all PeopleTools-owned object definitions that cannot be delivered using Copy Project to File.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-16-2: Load English Messages

This step runs the MSGTLENG.DMS script, which loads English messages into your database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-16-3: Load English String Data

This step runs the PTSTRENG.DMS script, which loads English string data into the STRINGS\_TBL table.

---

**Note.** The non-English language data was loaded in the task Loading Base Data.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-16-4: Load Stored Statements Data

Loading the stored statements ensures that the dynamic SQL statements will work correctly with the delivered COBOL programs.

This step runs the STOREPT.DMS script, which loads the dynamic SQL used by the PeopleTools-delivered COBOL.



### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-17: Creating PeopleTools Views

This section discusses:

- Create PPLTOOLS Views
- Create Updated PeopleTools Views

### Task 2-17-1: Create PPLTOOLS Views

This step creates all views defined in the PPLTOOLS project. These views are required for tasks later in the upgrade.

---

**Note.** The upgrade is designed to continue on regardless if any errors are encountered in this step. You may ignore any errors on views that fail to create at this time if they are created successfully in the next step “Create Updated PeopleTools Views.” However, if you see any other types of index creation errors, such as space problems, you must correct them before you continue with the upgrade.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 2-17-2: Create Updated PeopleTools Views

This step creates all views defined in the PPLTLS84CUR project. These are PeopleTools views that have changed and are required for tasks later in the upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-18: Converting PeopleTools Objects

This section discusses:

- Expand Fields on Records
- Alter Approval Rule Set Related Tables
- Update Message Nodes
- Update Message Nodes for Informix
- Convert Portal Objects
- Convert Query Prompt Headings
- Load Conversion Data
- Report Conversion Details
- Run Data Conversion

## Task 2-18-1: Expand Fields on Records

This step runs the Application Engine program, UPG84. This process clears the table PSRECFIELDDB. It then opens each record, expands the fields in subrecords, and writes the list of expanded fields into PSRECFIELDDB.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-18-2: Alter Approval Rule Set Related Tables

This step runs the Application Engine program VACONVERT2. It increases the size of the MAX\_DEBIT\_AMT and MAX\_CREDIT\_AMT columns in the PS\_APPR\_RULE\_AMT and PS\_APPR\_VA2\_WRK tables from DECIMAL (15, 2) to DECIMAL (26, 3). It also adds a column to the PSVAITEM table.

---

**Note.** When you modify tablespace and indexspace information, you also need to modify the VACONVERT2 program. DB2 z/OS customers also need to modify PRIQTY and SECQTY values.

---

See Appendix: “Modifying the Approval Rule Set Program.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-18-3: Update Message Nodes

This step runs the Data Mover script PORTALNODES.DMS. This script sets the default local node, the local default node’s portal and content URI text, node types, and the default portal name.

---

**Note.** If you are running on Informix, perform the next step instead.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Sybase Oracle Microsoft DB2 UNIX/NT DB2 z/OS	All

## Task 2-18-4: Update Message Nodes for Informix

This step runs the script PORTALNODES.SQL. This script sets the default local node, the local default node's portal and content URI text, node types, and the default portal name.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Informix	All

## Task 2-18-5: Convert Portal Objects

This step runs the UPG84PORTAL Application Engine program.

The UPG84PORTAL Application Engine program updates PeopleSoft portal objects to the new PeopleSoft format. You must have portal administration privileges when running this process.

The UPG84PORTAL process performs the following actions:

- Sets the node type to *EX* for message nodes whose URI doesn't contain *iclientservlet*.
- Sets the DEFAULT system tab dynamic categories.
- Converts registered portal content reference data to simple URL format if the node type for the cref is PIA or uses LOCAL\_NODE.
- Converts template content reference data, pagelet content reference EditURL data, and favorites data to simple URL format.
- Copies your home page personalization data and your favorites data to the default portal.
- Converts related links to simple URL format if the node type is set to Oracle's PeopleSoft Pure Internet Architecture.

You may see some errors or messages in your log. The following list contains some of the errors and what to do about them:

- Not authorized CRef: PT\_PORTAL\_HOMEPAGE\_GBL (95,5032).

This means that you do not have proper privileges to run this conversion. Grant the Portal Administrator role to the upgrade user ID.

- Security synchronization failed for Portal Object: *Portal Object Name*.

This is not a fatal error. It may be caused by a content reference that contains invalid URL text and indicates that there was an internal error writing to the security table. The invalid URL text may be pointing to a component or script that does not exist in the database. You need to fix the content reference and then the UPG84PORTAL process can be rerun.

- Duplicate key. Portal: *Portal Name*, Obj Name: *Portal Object Name*, Nodename: *Node*, URL: *URL* (133,4).

This portal object has the same URL as another portal object. Delete or modify this object to remove the conflict and then the UPG84PORTAL process can be rerun.

- Favorite not saved. Target content reference URL: *URL*, not found. Favorite: *favorite name*, user: *user name*. (95,5068)

Message Set Number: 95

Message Number: 5068

Message Reason: Favorite not saved. Target content reference URL: *URL*, not found. Favorite: *favorite name*, user: *user name*. (95,5068)

The favorite referenced in the previous message will not be converted to simple URL format. After the final Move to Production upgrade pass, the end-user will have to reenter the favorite with a valid value on the upgraded production database.

- Error getting Favorite for label XXXX.(Where XXXX represents a favorite label.) (96,58)

Ignore this error. It will not cause any problems to occur.

- Access denied, source message node not defined. (94,24)

Message Set Number: 94

Message Number: 24

Message Reason: Access denied, source message node not defined. (94,24)

The favorite referenced in this message will not be converted to simple URL format. After the final Move to Production upgrade pass, the end-user will have to reenter the favorite with a valid value on the upgraded production database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-18-6: Convert Query Prompt Headings

This step runs the Application Engine program UPGQRYDUPHED, which searches for duplicate prompt headings in the table PSQRYBIND and appends numbers onto the text. For example, *Item ID* would become *Item ID 2*. When you run Crystal 9 through the process scheduler, it cannot handle queries with two or more prompts that have the same heading. These duplicates are also not legal in Query. You need to alter any old queries that have duplicate prompt headings so that they work with Crystal 9.

If you find a duplicate heading that exceeds the length of the field **HEADING**, you need to change the heading manually. In these cases, the following error is written to the log file:

```
The prompt heading HEADING for Query QUERY is duplicated. Please manually correct.⇒
(108, 1108)
```

See the Enterprise PeopleTools PeopleBook: PeopleSoft Query for your new release.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-18-7: Load Conversion Data

This step imports PeopleTools data conversion Application Engine driver data into your database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-18-8: Report Conversion Details

This step runs the PTUCONV.SQR script. It details which sections will be called by the Upgrade Driver program and what they are doing. Each of the upgrade data conversion sections contains comments that describe the processing done by the section. The information contained in the report is used to evaluate the conversions run in the next step and any actions that are required as a result of the conversion.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 2-18-9: Run Data Conversion

The Upgrade Driver Application Engine program, PTUPGCONVERT, runs additional PeopleTools upgrade data conversions. The program then reads the table PS\_PTUPGCONVERT, selecting all rows with the group number of 01 and ordering them by the sequence number on the row. A list of Application Engine library sections that must be run for data conversion is returned. The program then calls each section in the order of the sequence number. After running PTUPGCONVERT, review the output data generated in the previous step for more details.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-19: Converting Integration Broker

This section discusses:

- Updating Integration Broker Defaults
- Creating Integration Broker Objects
- Saving Application Messaging Objects
- Exporting Node Transactions
- Preparing Integration Broker Deletes
- Deleting Application Messaging Objects
- Deleting Node Transactions

### Task 2-19-1: Updating Integration Broker Defaults

This step runs the PTIBUPGRADE.DMS script. This script populates the default values specified earlier in the upgrade.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 2-19-2: Creating Integration Broker Objects

The PeopleTools Upgrade Driver Application Engine program, PTUPGCONVERT, runs additional PeopleTools upgrade data conversions. The program then reads the table PS\_PTUPGCONVERT, selecting all rows with a group number of 03 and ordering them by the row sequence number. A list of Application Engine library sections that must be run for data conversion is returned. The program then calls each section in the sequence number order. Review the report generated by PTUCONV.SQR for details on the conversions run in this step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-19-3: Saving Application Messaging Objects

This step copies the PTUPGIBCLONE project to the *PS\_HOME*\projects directory. This project was created by the UPGPT848IBUG Application Engine program and contains objects that were successfully converted. The objects are copied to file as a precautionary measure because they will be deleted from the upgrade database.

After running this step, save the exported project in a permanent location where it can be accessed post-upgrade in case there is a need to review or import the old objects.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-19-4: Exporting Node Transactions

This step runs PTUPG\_TRX\_EXPORT.DMS to save out the old preconversion node transaction data. The generated .dat file is written to the DataMover output directory defined in Configuration Manager, which should be your *PS\_HOME*\data directory.

After running this step, save PTUPG\_TRX\_EXPORT.DAT in a permanent location where it can be accessed post-upgrade in case there is a need to review or import the old objects.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-19-5: Preparing Integration Broker Deletes

This step copies the PTUPGIBDELETE project to your *PS\_HOME*\projects directory in preparation for deleting the obsolete pre-conversion object definitions from the upgrade database. This project was created by the UPGPT848IBUG Application Engine program and contains the same objects as PTUPGIBCLONE.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-19-6: Deleting Application Messaging Objects

This step copies the PTUPGIBDELETE project definition from file. Since the actions in the project are set to Delete, this will delete the obsolete preconversion object definitions from the upgrade database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-19-7: Deleting Node Transactions

This step runs PTUPG\_TRX.DMS, which removes obsolete node transaction data associated with the obsolete objects in the PTUPGIBDELETE project. This script was generated by the UPGPT848IBUG Application Engine program.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 2-20: Verifying NonBase Language Pages

This section discusses:

- Understanding NonBase Language Pages
- Verify NonBase Language Pages
- Delete Unused Language Pages

### Understanding NonBase Language Pages

In this task, you review nonbase language pages created by the upgrade process and, if applicable, run a script to delete any unneeded nonbase language pages.

## Task 2-20-1: Verify NonBase Language Pages

In the new PeopleTools release, PeopleSoft has changed the way that translations of pages are stored. Instead of storing a complete copy of the page's structure per language, only the elements of a page that are language-sensitive, such as field labels and field position, are stored. All other elements of a page's design, such as the use properties of a field, the existence of a field, and any associated PeopleCode, are shared across all languages of a page.



Therefore, as part of your upgrade to the new PeopleTools release, the upgrade process scanned your nonbase language pages to determine if their structure is similar to the base language version of the same page. If the nonbase language version of a page is sufficiently similar to the base language version, PeopleTools combined these pages into one, and stored the language-sensitive elements of the page using the new architecture. However, if the nonbase language page does not match the structure of its base language equivalent, PeopleTools made a copy of the nonbase language page so you can maintain these changes. If you made customizations to nonbase language pages, you should review the contents of the PSX\_PAGEACTION table to determine which new pages were created by the upgrade process in order to preserve your changes. You can either add these newly created pages to menus, or delete the newly created pages and reapply your changes to the base language equivalents for these pages, which will then take effect across all languages for that page.

The following attributes of a page were compared against the same attributes of each associated nonbase language page during execution of the REL script:

- Number of fields
- For each equivalent field on the base and nonbase pages, whether any of the following do not match:
  - Field Order
  - Panel Field Name (PNLFIELDNAME)
  - Record Name
  - Record Field Name
  - Occurs Level
  - Field Type (for example, checkbox, dropdown list, or edit box)
  - Field Use (for example, invisible or related display) except for Use Default Label

If any of these attributes differed between the base language page and its nonbase language equivalents, the nonbase language page was duplicated. If all these attributes matched, then the base and nonbase language pages differed only in user interface and were merged.

Follow the steps below to determine which action was taken against each of your pages and the appropriate response.

To verify nonbase language pages:

1. Examine the PSPNLDEFN with the following SQL statement. The results will be the original page name plus underscore language code (for example, TRANSLATION\_RECORDS\_FRA), and the name of the new page created to preserve the language-specific structural changes.

```
SELECT DESCR, PNLNAME
FROM PSPNLDEFN
WHERE PNLNAME LIKE 'FIXPGNAME%'
```

2. Take one of the following actions on the pages from the output of the SQL select statement above:
  - If you did not customize the pages, or if you no longer want to maintain the structural changes on the nonbase language page that differ from the base language page, you can safely delete the newly created pages using Application Designer, once you perform a backup after upgrading.

See Backing Up After PeopleTools Upgrade.

- If you do not want to preserve the language-specific structural changes to any of the pages in the result set from the SQL select statement above, follow the instructions in the next step to delete all corresponding duplicate pages.

See Delete Unused Language Pages.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-20-2: Delete Unused Language Pages

If you currently use only the base language of your database or do not want to preserve any of the customizations you made to nonbase language pages, you can delete all of the new pages created by the upgrade process (where nonbase languages did not match their base language equivalents). Run the following script manually in Data Mover user mode to maintain only the customizations on the base language pages and to delete customizations on nonbase language pages.

```
PT84RLPD.DMS
```

If you want to maintain customizations on any nonbase language pages and preserve any of the duplicate pages, do not run this script.

---

**Important!** The PSX\_PAGEACTION table will be deleted later in the upgrade. Complete this task and record the contents of PSX\_PAGEACTION before proceeding with your upgrade. The record PSX\_PAGEACTION contains the pages that have been renamed to *FIXPGNAME%*. For pages with a CONVERT\_ACTION of *X*, the structure of the pages matched, and were merged. No duplicate was created. You do not need to perform any action for these pages. For pages with a CONVERT\_ACTION of *D*, the nonbase language version of the page did not match the structure of the equivalent base language page and was duplicated (for example, FIXPGNAME1).

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 2-21: Clearing Process and Report Tables

This section discusses:

- Understanding Process and Report Tables
- Purge Process Request Tables

## Understanding Process and Report Tables

In the new PeopleTools release, some new Process Request tables were introduced and are used when scheduling a request or job. After upgrading from a previous PeopleTools release, the content of these tables will be inconsistent with how the Process Scheduler modules use them. This will cause unexpected results when viewing the status of these requests from the Process Monitor page. You must clear the content of these tables before scheduling any requests.

### Task 2-21-1: Purge Process Request Tables

This step runs the PRCSCLR.DMS script, which purges all entries found in the Process Request tables.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-22: Setting Object Version Numbers

In this task, you run the VERSION Application Engine program. This ensures that all of your version numbers are correct and, if not, resets them to 1.

**Note.** You will rerun the VERSION application engine program later in the upgrade. If you want to preserve the log files generated by Change Assistant from this run, you will need to rename the files manually after completing this task.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 2-23: Converting Database Data Types

This section discusses:

- Understanding Converting Database Data Types
- Backing Up Before Platform Changes
- Running Long Data Audit
- Validating Microsoft Database
- Reviewing Microsoft Settings

- Creating Microsoft Conversion Project
- Generating Microsoft Conversion Script
- Running Microsoft Conversion Script
- Granting Permissions to the CONNECT ID
- Running Microsoft Conversion Report
- Validating Oracle Database
- Creating Oracle Audit Tables
- Auditing Duplicate Length Constraints
- Auditing Disabled Constraints
- Reviewing Oracle Settings
- Generating Oracle Conversion Scripts
- Running Long to LOB Script 1
- Running Long to LOB Script 2
- Running Long to LOB Script 3
- Running Long to LOB Script 4
- Running Long to LOB Script 5
- Running Long to LOB Script 6
- Running Long to LOB Script 7
- Running Long to LOB Script 8
- Auditing Long to LOB Conversion
- Running Character Length Script 1
- Running Character Length Script 2
- Running Character Length Script 3
- Running Character Length Script 4
- Running Character Length Script 5
- Running Character Length Script 6
- Running Character Length Script 7
- Running Character Length Script 8
- Auditing Character Length Semantics
- Reviewing Conversion Reports
- Updating Database Options

## Understanding Converting Database Data Types

As of PeopleTools 8.48, new database data types are supported for Microsoft SQL Server 2005 or later and Oracle 9i or later. These data type changes are mandatory but only available for use in conjunction with application releases 9.0 or later.

For Microsoft SQL Server 2005 and later, the data types VARCHAR, NVARCHAR, VARBINARY(MAX), and VARCHAR(MAX) are now supported. Databases on Microsoft SQL Server 2000 and earlier will not use these new data types. The data types as defined in Application Designer are not changed; only the database-level definition will be different:

- Records with fields defined as PeopleSoft CHAR(N) will now use VARCHAR(N).
- Records with fields defined as PeopleSoft NCHAR(N) will now use NVARCHAR(N).
- Records with fields defined as PeopleSoft Long Character(N) will now use VARCHAR(N) if N is <=4000 and VARCHAR(MAX) if N is > 4000 for non-Unicode.
- Records with fields defined as PeopleSoft Long Character(N) will now use NVARCHAR(N) if N is <=4000 and VARCHAR(MAX) if N is > 4000 for Unicode databases.
- Records with fields defined as PeopleSoft IMAGE will now use VARBINARY(MAX).

For Oracle 9i or later, the data types CLOB and BLOB are now supported. In addition, the Character Length Semantics feature is also supported for Unicode databases when creating PeopleSoft CHAR fields and LONG CHARACTER fields with specified lengths less than 1334:

- Records with fields defined as PeopleSoft IMAGE or PeopleSoft LONG CHARACTER with Raw Binary will now use BLOB.
- Records with fields defined as PeopleSoft LONG CHARACTER with no length specified, length greater than 1333(UNICODE), or length greater than 4000 (ANSI) will now use CLOB.

## Task 2-23-1: Backing Up Before Platform Changes

Back up your upgrade database now. This enables you to restart your upgrade from this point, in case you experience any database integrity problems during the remaining tasks in the upgrade process.

---

**Important!** For Oracle platforms, contact your database administrator to update the statistics on the database catalog. This will improve performance for subsequent steps in the upgrade. Typically only the users sys and sysdba have the authority to perform this task.

The following command updates the statistics on the database catalog:

```
EXEC DBMS_STATS.GATHER_SCHEMA_STATS ( 'SYS' ) ;
```

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server Oracle	All

## Task 2-23-2: Running Long Data Audit

This step runs LONGS-AUDIT.SQL, which audits for any fields exceeding the actual data length for PeopleSoft long character columns. You will review the output in a later step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-3: Validating Microsoft Database

This step runs DBSETTINGS.SQL, which checks the Microsoft SQL Server version. The data type conversion is supported only with Microsoft SQL Server 2005 or later. You will review the output in a later step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-4: Reviewing Microsoft Settings

If you are upgrading to a PeopleSoft 9.0 or later application release, the data type update *and* a minimum of Microsoft SQL Server 2005 are required. You will run a conversion process that will substitute the old data types for new ones. The data type conversion is supported for Microsoft SQL Server 2005 or later with PeopleTools 8.48 or later and an application release 9.0 or later. Examine the log file from the step Validating Microsoft Database to ensure that you are running a supported version of Microsoft SQL Server. Do *not* perform the rest of this task if you do not meet the qualifications.

Examine the log file from the step Running Long Data Audit to determine if there are any fields shorter than length 4000 in the database that exceed the actual data length defined for the PeopleSoft long character fields. Prior to PeopleTools 8.48, all PeopleSoft long character fields were created using the TEXT SQL Server data type, and no matter the length defined by the Application Designer, the data in the field could grow as much as the TEXT limits on SQL Server. After the data type conversion, the length specified in Application Designer will be enforced for all fields shorter than length 4000, except for those with length zero. If your data is larger than the length defined in Application Designer, then you must correct the length using Application Designer or change the data itself using your SQL query tool. You must decide whether you want a change in the field length definition or a change in the data. The log file created by LONGS-AUDIT.SQL will only show all of the fields that contain data exceeding a length between 1 and 4000 and will be empty if this condition does not occur with no other action to take.

Resolve these problems before continuing to the next step, otherwise the conversion process will fail. If necessary, contact your DBA for assistance in modifying the fields. If no fields are listed in the log file, no further action is needed and you may proceed with the upgrade.

---

**Note.** During Move to Production passes, copy MSSNEWTYPE\_ALTER.SQL from your initial pass upgrade's output directory and place it into the output directory for your Move to Production pass. This script is only generated during the initial pass. Edit the script and correct the database name on the first line of the script to point to the Target database for the pass.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-5: Creating Microsoft Conversion Project

This step runs MSSNEWTYPE.SQL, which generates and populates the MSSNEWTYPE project. The project contains all of the records that need to be modified to use the newly supported data types.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	MS SQL Server	All

## Task 2-23-6: Generating Microsoft Conversion Script

This step generates the SQL script MSSNEWTYPE\_ALTER.SQL to alter the records in the MSSNEWTYPE project. The generated script will alter the tables with the new data types.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	MS SQL Server	All

## Task 2-23-7: Running Microsoft Conversion Script

This step runs the generated script from the previous step. This will alter the existing tables to use the new data types. All of the tables will be copied into their new representation using the new data types and all of the additional padding blanks derived from the use of the old data types will be truncated.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-8: Granting Permissions to the CONNECT ID

This step runs the GRANT.SQL script. This script grants select access to the CONNECT ID for tables necessary for sign on.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-9: Running Microsoft Conversion Report

This step runs CONVERSION-AUDIT.SQL, which audits for all unconverted fields. You will review the output in a later step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

## Task 2-23-10: Validating Oracle Database

This step runs the DBSETTINGS.SQL script, which queries the database to determine the value of the NLS\_LENGTH\_SEMANTICS parameter. You will review the output in a later step.

There are two possible conversions that may occur depending on whether or not the database is Unicode. The Long to LOB conversion will apply to all databases, Unicode or ANSI. CHARACTER LENGTH SEMANTICS (CLS) only applies to Unicode databases. The CLS conversion has a dependency on the init.ora parameter NLS\_LENGTH\_SEMANTICS. The init.ora parameter NLS\_LENGTH\_SEMANTICS=CHAR, must be enabled for PeopleSoft Enterprise Unicode databases prior to executing the conversion. If the database being converted is ANSI, then this setting is not necessary.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-11: Creating Oracle Audit Tables

This step runs PRECNVADT1A.SQL, which drops and re-creates some temporary tables required by the pre-conversion audit SQRs.

If the tables being dropped, CHECK\_CONSTRAINTS, DUPLICATE\_CONSTRAINTS, and DROP\_CONSTRAINTS, don't exist, the execution of this script will generate the following error, which can safely be ignored:

```
ORA-00942: table or view does not exist
```



## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-12: Auditing Duplicate Length Constraints

This step runs PRECNVADT1.SQR, which checks for duplicate length constraints. This condition can generally exist if the database was created using the Oracle Import utility and CONSTRAINTS=Y was enabled, which is the default setting. You will review the output in a later step.

---

**Note.** If this SQR needs to be rerun for any reason, you *must* run PRECNVADT1A.SQL before rerunning PRECNVADT1.SQR.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-13: Auditing Disabled Constraints

This step runs PRECNVADT2.SQR, which checks for ‘not\_validated’ constraints. Although this condition should not exist in a production database, it may have occurred if data was imported with external utilities, such as SQL Loader. You will review the output in a later step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-14: Reviewing Oracle Settings

The data type conversion is only supported for Oracle 9i or later with PeopleTools 8.48 or later *and* an application release 9.0 or later. Do *not* perform the rest of this task if you do not meet the qualifications.

For Unicode databases, examine the log file from the step Auditing Duplicate Length Constraints. If there are any duplicate length constraints, those duplicate constraints must be dropped. Run the utility SQL script, *PS\_HOME\scripts\GENDROPDUPCONSTRAINTS.SQL*, to generate the script *DROPDUPCONSTRAINTS.SQL*, containing an *ALTER TABLE TABLE\_NAME DROP CONSTRAINT* for every duplicate constraint found. Run the *DROPDUPCONSTRAINTS.SQL* to resolve the duplicate length constraints.

For Unicode databases, examine the log file from the step Auditing Disabled Constraints. If there are any disabled or nonvalidated constraints, these constraints should be re-validated. Run the utility SQL script, *PS\_HOME\scripts\GENREVALIDATECONSTRAINTS.SQL* to generate the script *REVALIDATECONSTRAINTS.SQL*, containing an *ALTER TABLE TABLE\_NAME ENABLE VALIDATE CONSTRAINT CONSTRAINT\_NAME* for every invalid constraint found. Run the *REVALIDATECONSTRAINTS.SQL* to enable the constraints.

For Unicode databases, examine the log file from the step Validating Oracle Database to determine if the values in the *init.ora* file are set properly. For Unicode databases, the *NLS\_LENGTH\_SEMANTICS* parameter needs to have a value of *CHAR*. This indicates that *CHARACTER\_LENGTH\_SEMANTICS* is enabled and the conversion can continue. If you need to enable Character Length Semantics, work with your DBA to modify the *init.ora* for the Target database's SID and set *NLS\_LENGTH\_SEMANTICS* to *CHAR*. Then stop and restart the database SID for the setting to take effect.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-15: Generating Oracle Conversion Scripts

Work with your database administrator to set the following *init.ora* parameters for the Target database's system identifier (SID). Stop and restart the database SID for the following settings to take effect:

1. Set the following *init.ora* parameters:

```
db_block_size=8192
db_cache_size=325165824
db_file_multiblock_read_count=8
job_queue_processes=10
shared_pool_size=425829120
pga_aggregate_target=5871947670
parallel_max_servers=8
workarea_size_policy=AUTO
```

---

**Note.** If you are using Oracle 10g or higher, you may use the parameters *SGA\_TARGET=300M* and *SGA\_MAX\_SIZE=350M* instead of *SHARED\_POOL\_SIZE*, *DB\_CACHE\_SIZE*, and *DB\_BLOCK\_BUFFERS*.

---

2. Pre-allocate the *PSTEMP* tablespace to at least 10 GB.
3. Pre-allocate the *PSDEFAULT* tablespace to at least 2 GB with 10-MB local uniform extents.
4. Ensure that you have at least six redo logs sized at 500 MB each.

The Oracle data types script generation program is a Java program that connects to an Oracle database. The prerequisites are Java and the Oracle JDBC Drivers. The Oracle JDBC drivers will automatically be picked up by the *.bat* file provided that the *%ORACLE\_HOME%* environment variable is set. Check if the *%ORACLE\_HOME%* environment variable is set, by typing the following at the workstation command prompt:

```
echo %ORACLE_HOME%;
```

This should return a path, for example:

```
c:\oracle\product\10.1.0\client_1;
```

If the %ORACLE\_HOME% environment variable is not set, then set it in the command prompt window by typing the following at the workstation command prompt:

```
SET ORACLE_HOME=Oracle_Home_location
```

The Oracle data types script generation program is executed using the *PS\_HOME*\utility\PSORADDataTypesConversion.BAT file, which requires five input parameters:

- **THREADS**: The number of Java threads the conversion script generation spawns to generate the scripts. We recommend 10 threads for running this program on Windows.
- **ACCESSID**: The access ID for the database to be converted.
- **ACCESSIDPW**: The access password for the database to be converted.
- **DBNAME**: The database name.
- **OUTPUTDIR**: A directory path to redirect the generated conversion scripts to a user specified directory. This must be set to the Change Assistant output directory for your upgrade pass. Change Assistant will run the generated scripts later in the upgrade.

Example:

```
PS_HOME\utility\PSORADDataTypesConversion.bat 10 SYSADM SYSADM MYDB c:\upgrade=>
\output\Change_Assistant_job_directory
```

In the example command line above:

```
THREADS = 10
```

```
ACCESSID = SYSADM
```

```
ACCESSIDPW = SYSADM
```

```
DBNAME = MYDB
```

```
OUTPUTDIR = c:\upgrade\output\Change_Assistant_job_directory
```

Open a command prompt window on the client workstation and execute the Oracle data types script generation program *PS\_HOME*\utility\PSORADDataTypesConversion.bat. The program will display and write a log (PsOraCnv.log) to the directory specified by the OUTPUTDIR parameter indicating the status of the conversion program. Review PsOraCnv.log and ensure that the conversion scripts were generated cleanly.

For ANSI databases, only LONGTOLOBALTER conversion scripts are generated. For Unicode databases, two sets of scripts are generated: LONGTOLOBALTER conversion scripts and CHARACTERLENGTHSEMANTICSALTER scripts.

After successfully running the conversion program, verify that the generated SQL scripts are located in the staging Change Assistant output directory for your upgrade pass. Later in the upgrade, Change Assistant will automatically run the SQL scripts later in the upgrade from the Change Assistant output directory for your upgrade pass.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-16: Running Long to LOB Script 1**

This step runs LONGTOLOBALTER1.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-17: Running Long to LOB Script 2**

This step runs LONGTOLOBALTER2.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-18: Running Long to LOB Script 3**

This step runs LONGTOLOBALTER3.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-19: Running Long to LOB Script 4**

This step runs LONGTOLOBALTER4.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-20: Running Long to LOB Script 5**

This step runs LONGTOLOBALTER5.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-21: Running Long to LOB Script 6**

This step runs LONGTOLOBALTER6.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-22: Running Long to LOB Script 7**

This step runs LONGTOLOBALTER7.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-23: Running Long to LOB Script 8**

This step runs LONGTOLOBALTER8.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle long to LOB conversion scripts are designed to run concurrently to improve performance.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-24: Auditing Long to LOB Conversion

This step runs L2LAUDIT.SQR to report on the output of the long to LOB conversion. You will review the report output in a later step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-25: Running Character Length Script 1

This step runs CHARACTERLENGTHSEMANTICSALTER1.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-26: Running Character Length Script 2

This step runs CHARACTERLENGTHSEMANTICSALTER2.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-27: Running Character Length Script 3

This step runs CHARACTERLENGTHSEMANTICSALTER3.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-28: Running Character Length Script 4**

This step runs CHARACTERLENGTHSEMANTICSALTER4.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-29: Running Character Length Script 5**

This step runs CHARACTERLENGTHSEMANTICSALTER5.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-30: Running Character Length Script 6**

This step runs CHARACTERLENGTHSEMANTICSALTER6.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 2-23-31: Running Character Length Script 7**

This step runs CHARACTERLENGTHSEMANTICSALTER7.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-32: Running Character Length Script 8

This step runs CHARACTERLENGTHSEMANTICSALTER8.SQL, which was generated using PSORADDataTypesConversion.bat. The Oracle character length semantics conversion scripts are designed to run concurrently to improve performance.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-33: Auditing Character Length Semantics

This step runs CLSAUDIT.SQR to report on the output of the character length semantics conversion. You will review the report output in a later step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 2-23-34: Reviewing Conversion Reports

To review the conversion report for Microsoft, examine the log file from the step Running Microsoft Conversion Report. It contains a list of unconverted columns on tables along with its old data type. Fields on tables with no Application Designer definition will be included in this log. Any unresolved errors from the step Running Microsoft Conversion will also be included. If you are using these tables it is possible to update them manually to use the new data types with a SQL query tool or with an ETL tool. Be very cautious when changing a table as this could result in data loss or affected functionality. Once any underlying problems have been resolved, you may rerun all of the previous steps in this task to reconvert any remaining objects listed by the audit report.

---

**Note.** During Move to Production passes for Microsoft, you must manually convert any remaining objects. During Move to Production passes, the record definition differs from the database table structure, so do *not* build the record with Application Designer.

---



To review the conversion reports for Oracle, examine the log files from running the LONGTOLOBALTER\*.SQL scripts. If the database is Unicode, also examine the log files for the CHARACTERLENGTHSEMANTICS\*.SQL scripts. Review the output from the step “Auditing Long to LOB Conversion.” L2LAUDIT.SQR reports on any unconverted long raw columns. The tablename, columnname, and column data type are listed. For unicode databases, review the output from the step “Auditing Character Length Semantics.” CLSAUDIT.SQR reports on any unconverted character length columns (Unicode only). Correct any errors listed on the log files or conversion reports before proceeding with the upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server Oracle	All

## Task 2-23-35: Updating Database Options

This step runs UPGDBOPTIONS\_ENABLE.SQL. This script updates the database to indicate that the new data types are now enabled.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server Oracle	All

---

## Task 2-24: Backing Up After PeopleTools Upgrade

Back up your Copy of Production database now. This enables you to restart your upgrade from this point, in case you experience any database integrity problems during the remaining tasks in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All



## CHAPTER 3

# Run and Review Compare Reports

This chapter discusses:

- Understanding Compare Reports
- Preparing for Application Changes
- Renaming Tables
- Running New Release Compare Reports
- Reviewing New Release Compare Reports

---

## Understanding Compare Reports

Now that your Copy of Production database is at the same PeopleTools release as your new release, you can compare the two databases to see the differences. In this chapter you run and review compare reports to make decisions regarding your upgrade. Be sure you have plenty of space to run these reports, as some can be rather large.

---

## Task 3-1: Preparing for Application Changes

This section discusses:

- Running Drop Table Script
- Loading HR Ethnicities Table
- Exporting Project Definitions
- Importing Project Definitions
- Dropping View ADP\_ACCT\_CD\_VW
- Running the Personal Data Setup Script
- Renaming Fields for Microsoft SQL Server

### Task 3-1-1: Running Drop Table Script

In this step you run the drop table script. It will drop tables that contain data that is no longer needed or cannot be converted.

The script name for your upgrade is:

DLHCUPG01.DMS

If you want to make this step automated, follow the steps below.

To make this step automated:

1. Select the step Running Drop Table Script in Change Assistant.
2. Open the Step Properties dialog box.
3. Change the Type from *ManualStop* to *DataMoverUser*.
4. Click OK.
5. In your upgrade job, mark the step as Run.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 3-1-2: Loading HR Ethnicities Table

In this step you populate the UPG\_HR\_ETHNIC table for upgrading ethnicities. You will use this data during data conversion. The DataMover script for your path is:

DLHCHRW01.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 3-1-3: Exporting Project Definitions

In this step you will export from your Demo database the definition of projects that will be used later in this upgrade. You will import these definitions in the next step. Your export script is:

DLUPX08E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 3-1-4: Importing Project Definitions

In this step you will import the project definitions into your Copy of Production database. These projects will be used later in this upgrade. Your import script is:

DLUPX08I.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 3-1-5: Dropping View ADP\_ACCT\_CD\_VW

Run this task only if you have ADP Connector installed and DB2 z/OS or DB2 UDB database.

This step drops the view ADP\_ACCT\_CD\_VW to avoid possible errors when renaming table ACCT\_CD\_TBL in the task Renaming Tables. Run the following script to drop the view:

```
DLHCPIW01.DMS
```

If you want to automate this step, follow the procedure below.

To make this step automated:

1. Select the step Dropping View ADP\_ACCT\_CD\_VW in Change Assistant.
2. Open the Step Properties dialog box.
3. Change the Type from *ManualStop* to *DataMoverUser*.
4. Click OK.
5. In your upgrade job, mark the step as Run.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll Interface	DB2 z/OS DB2/UNIX NT	All

## Task 3-1-6: Running the Personal Data Setup Script

This script replaces references on record and field definitions of PERSONAL\_DATA with PERSON or PERSON\_NAME where needed. This will greatly reduce the reported number of differences in the record compare.

Run the following script:

```
DLHCHRG10.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 3-1-7: Renaming Fields for Microsoft SQL Server

Earlier in the upgrade, you reversed the rename of fields to be deleted in the new release so that they would not be altered during the PeopleTools data type conversion. This script will reapply those renames so that your compare to the New Release Demo database is consistent with your earlier compare.

Run the following script:

RNHCUPG05.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	MS SQL Server	All

---

## Task 3-2: Renaming Tables

This section discusses:

- Understanding Renamed Tables
- Running the RNHCUPG03MSS Script
- Running the RNHCUPG03DB2 Script
- Running the RNHCUPG03DBX Script
- Running the RNHCUPG03IFX Script
- Running the RNHCUPG03ORA Script
- Running the RNHCUPG03SYB Script
- Running the RNCSUPR04MSS Script
- Running the RNCSUPR04DB2 Script
- Running the RNCSUPR04DBX Script
- Running the RNCSUPR04IFX Script
- Running the RNCSUPR04ORA Script
- Running the RNCSUPR04SYB Script
- Running the RNCSUPG02MSS Script
- Running the RNCSUPG02DB2 Script
- Running the RNCSUPG02DBX Script
- Running the RNCSUPG02IFX Script
- Running the RNCSUPG02ORA Script
- Running the RNCSUPG02SYB Script

## Understanding Renamed Tables

These SQL scripts rename tables, at the database level, to temporary table names. They do not change the Record Definition. These temporary tables will be used in the data conversion programs in a later step.

Near the end of the upgrade tasks, you will run a DDDAUDIT report again. On the report, these temporary tables will be listed in the section listing: “SQL Table defined in the Database and not found in the Application Designer.” Either at that point or later when you are comfortable with the results of the data conversion, you can drop these temporary tables.

In some database platforms, the related indexes and views must be dropped before the table can be renamed. Oracle has included drop statements for these objects that exist on the DMO version of the database. However, the list of related objects may be different in your environment because of customizations or applied product incidents. You may encounter errors in these scripts because of these differences—for example, the script tries to drop an index that you do not have or it cannot rename the table because there are more related objects that need to be dropped. Simply modify these scripts to work for your database and you will not encounter these errors in your next test pass.

### Task 3-2-1: Running the RNHCUPG03MSS Script

The RNHCUPG03MSS.SQL script renames tables on the Copy of Production database. This script is for SQL Server databases and runs in the initial and Move to Production passes.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

### Task 3-2-2: Running the RNHCUPG03DB2 Script

The RNHCUPG03DB2.SQL script renames tables on the Copy of Production database. This script is for DB2 z/OS databases and runs in the initial and Move to Production passes.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

### Task 3-2-3: Running the RNHCUPG03DBX Script

The RNHCUPG03DBX.SQL script renames tables on the Copy of Production database. This script is for DB2 UDB databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

**Task 3-2-4: Running the RNHCUPG03IFX Script**

The RNHCUPG03IFX.SQL script renames tables on the Copy of Production database. This script is for Informix databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

**Task 3-2-5: Running the RNHCUPG03ORA Script**

The RNHCUPG03ORA.SQL script renames tables on the Copy of Production database. This script is for Oracle databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 3-2-6: Running the RNHCUPG03SYB Script**

The RNHCUPG03SYB.SQL script renames tables on the Copy of Production database. This script is for Sybase databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Sybase	All

**Task 3-2-7: Running the RNCSUPR04MSS Script**

The RNCSUPR04MSS.SQL script renames tables on the Copy of Production database. This script is for SQL Server databases and runs in the initial and Move to Production passes.



**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	MS SQL Server	All

**Task 3-2-8: Running the RNCSUPR04DB2 Script**

The RNCSUPR04DB2.SQL script renames tables on the Copy of Production database. This script is for DB2 z/OS databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	DB2 z/OS	All

**Task 3-2-9: Running the RNCSUPR04DBX Script**

The RNCSUPR04DBX.SQL script renames tables on the Copy of Production database. This script is for DB2 UDB databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	DB2 UNIX/NT	All

**Task 3-2-10: Running the RNCSUPR04IFX Script**

The RNCSUPR04IFX.SQL script renames tables on the Copy of Production database. This script is for Informix databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	Informix	All

**Task 3-2-11: Running the RNCSUPR04ORA Script**

The RNCSUPR04ORA.SQL script renames tables on the Copy of Production database. This script is for Oracle databases and runs in the initial and Move to Production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	Oracle	All

### Task 3-2-12: Running the RNCSUPR04SYB Script

The RNCSUPR04SYB.SQL script renames tables on the Copy of Production database. This script is for Sybase databases and runs in the initial and Move to Production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	Sybase	All

### Task 3-2-13: Running the RNCSUPG02MSS Script

The RNCSUPG02MSS.SQL script renames tables on the Copy of Production database. This script is for SQL Server databases and runs in the initial and Move to Production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	MS SQL Server	All

### Task 3-2-14: Running the RNCSUPG02DB2 Script

The RNCSUPG02DB2.SQL script renames tables on the Copy of Production database. This script is for DB2 z/OS databases and runs in the initial and Move to Production passes.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

### Task 3-2-15: Running the RNCSUPG02DBX Script

The RNCSUPG02DBX.SQL script renames tables on the Copy of Production database. This script is for DB2 UDB databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

**Task 3-2-16: Running the RNCSUPG02IFX Script**

The RNCSUPG02IFX.SQL script renames tables on the Copy of Production database. This script is for Informix databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

**Task 3-2-17: Running the RNCSUPG02ORA Script**

The RNCSUPG02ORA.SQL script renames tables on the Copy of Production database. This script is for Oracle databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

**Task 3-2-18: Running the RNCSUPG02SYB Script**

The RNCSUPG02SYB.SQL script renames tables on the Copy of Production database. This script is for Sybase databases and runs in the initial and Move to Production passes.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Sybase	All

---

**Task 3-3: Running New Release Compare Reports**

This section discusses:

- Understanding the New Release Compare
- Preserving Local Message Node

- Modifying New Release Compare Options
- Running New Release UPGCUST Compare
- Running New Release UPGCUSTIB Compare
- Resetting Take Action Flags in UPGCUST
- Running UPGCUSTIB Filter Script

## Understanding the New Release Compare

In this task you will compare your customizations to the new release objects by running a project compare against the Demo database.

### Task 3-3-1: Preserving Local Message Node

In this step add your Local Message Node to the UPGCUST project before the project compare between the Copy of Production and Demo database.

To add Local Message Node:

1. Run the following SQL in your Copy of Production database to identify your Local Message Node:

```
SELECT MSGNODENAME FROM PMSGNODEDEFN WHERE LOCALDEFAULTFLG = 'Y'
```

2. Open Application Designer from your Copy of Production database.
3. Open project UPGCUST.
4. Select Insert, Definitions Into Project.
5. Select Definition Type of Message Node.
6. Enter the Message Node name that was returned from the SQL you ran above.
7. Select the displayed Message Node.
8. Click Insert.
9. Save project UPGCUST.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

### Task 3-3-2: Modifying New Release Compare Options

For compare steps, PeopleSoft Change Assistant templates are delivered with the default reports filter turned on in the compare options. This limits the size of the reports and keeps them manageable. Before you start the compares, review the Change Assistant template step properties for each compare step in this task and modify the compare options based on your requirements.

If you decide not to modify the compare options, the objects are still compared. However, the results are only available online in Application Designer. They are not written to the compare reports. The compare reports are tools to help you review changed objects. However, based on the report filters you select, you may need to review the action flags for other objects in the compare project in Application Designer.

For example, you can modify the compare options so the report contains customized objects that are present in your Copy of Production database but absent from the Demo database. Alternatively, you can review these objects online, through Application Designer, after the compare.

To modify upgrade compare options:

1. Highlight the compare step you want to review and right-click.
2. Select Step Properties.

The Step Properties dialog box appears.

3. Click Upgrade.

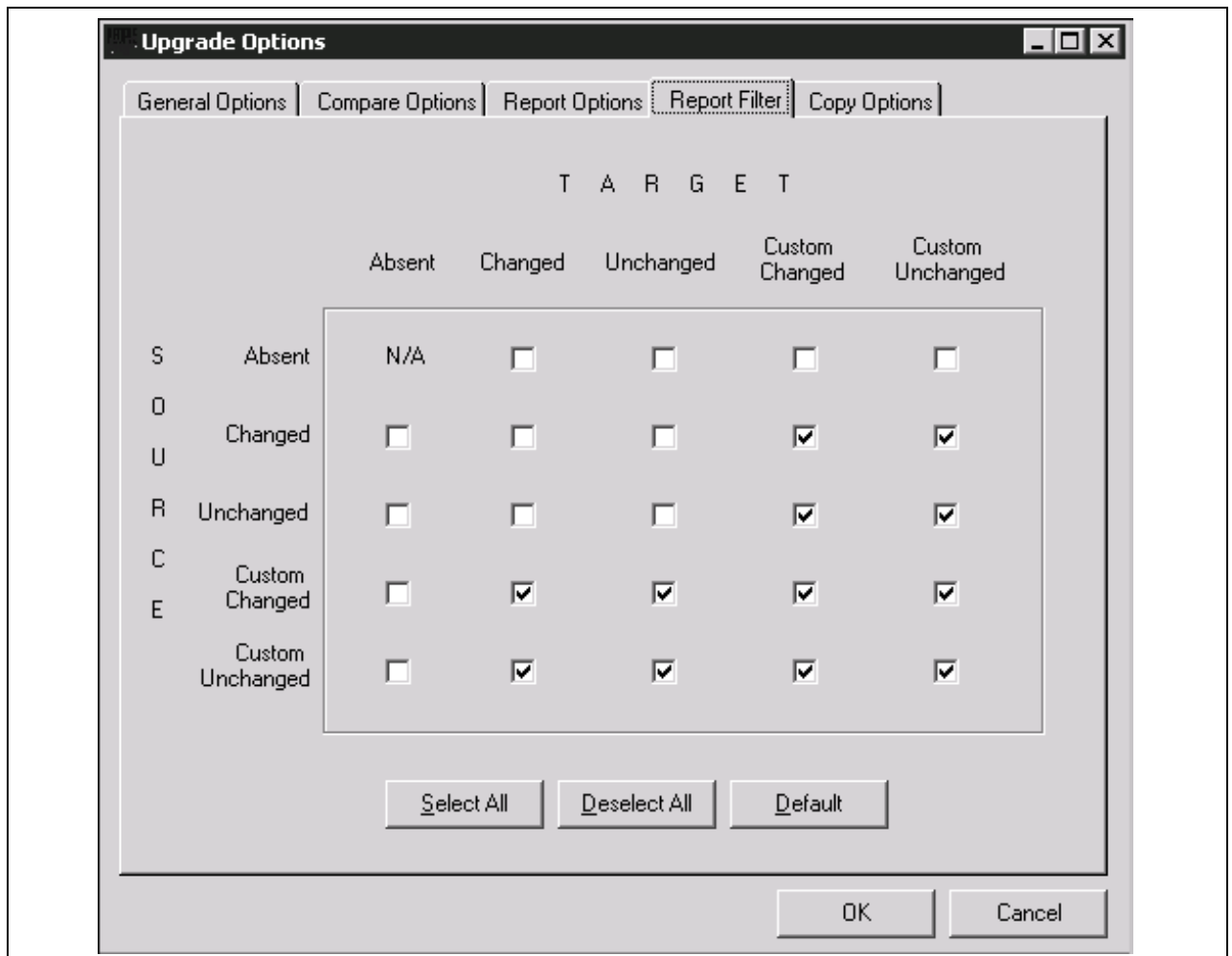
The Compare and Report dialog box appears.

4. Click Options.

5. Select the Report Filter tab.

The default options include your custom changes on the reports.

6. Change the default options as necessary and click OK.



Upgrade Options page, Report Filter tab

7. In the Compare and Report dialog box, click OK.
8. In the Step Definitions dialog box, click OK.

9. Repeat steps 1 through 8 for each compare step in the task you want to review and change.
10. Select File, Save Job.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 3-3-3: Running New Release UPGCUST Compare

This step executes a project compare of all comparable objects in the UPGCUST project.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 3-3-4: Running New Release UPGCUSTIB Compare

This step executes a database compare of all Integration Broker objects and generates the UPGCUSTIB project.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 3-3-5: Resetting Take Action Flags in UPGCUST

This step turns *off* all Take Action flags in the UPGCUST project after the compare. You will analyze the compare results and adjust the upgrade flags in the next step.

The script for your upgrade is:

```
DLUPX98.DMS
```

See Appendix: “Using the Comparison Process.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 3-3-6: Running UPGCUSTIB Filter Script

This step removes all objects from the UPGCUSTIB project that are not marked *\*Changed* or *\*Unchanged* in your Copy of Production environment. It is used to isolate only custom objects in the UPGCUSTIB project.

The script name for your upgrade is:

```
DLUPX95.DMS
```

See Appendix: “Using the Comparison Process.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 3-4: Reviewing New Release Compare Reports

This section discusses:

- Reviewing New Release Changes
- Reviewing NonComparable Objects

### Task 3-4-1: Reviewing New Release Changes

In this step, review the UPGCUST and UPGCUSTIB projects to evaluate how changes in the new release affect your customizations.

The UPGCUST and UPGCUSTIB projects exist in your Copy of Production database and contain all the objects you customized in the old release. These projects may include object definitions that are on your Copy of Production database but are not on the Copy of Current Demo database. If these are custom objects that you intend to keep in your upgraded system, you will want to ensure they are set to copy in the UPGCUST or UPGCUSTIB project. Compare reports are viewable when you open the projects in Application Designer. You can use these reports to determine your copy action for each object in the projects. Currently all Upgrade Flags are deselected, meaning no action will take place. Analyze the UPGCUST and UPGCUSTIB projects and select the Upgrade Flags for the customizations you wish to retain.

If the Target column has the value *Absent* it can indicate one of two possible conditions. If Oracle originally delivered the object definition, then it can be considered obsolete in the new release. This value can also indicate that you originally created the object definition for some custom functionality. To ensure the integrity and functionality of the system, delete obsolete Oracle-delivered objects. If you have made a customization to an obsolete object, refer to the product’s Release Notes to assess the functionality of the customization and determine where to reapply it in the new release.

See Appendix: “Using the Comparison Process.”

---

**Warning!** Pay close attention to the compare results for URLs, permission lists, and message nodes. It is highly likely that you will want to keep any customizations that you have made to these objects. You will want to migrate your customized local message node. Please be sure to select the Upgrade Flags from within Application Designer to retain these customizations.

---



---

**Note.** Steps in the database or third-party software installation documentation can result in Oracle-delivered objects being identified in the compare reports as *\*Changed* in the Source column. You should investigate all instances where objects are identified as *\*Changed* in the Source column to determine their origin and determine a plan of action based on the findings for each object.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 3-4-2: Reviewing NonComparable Objects

The UPGNONCOMP project is delivered in your Demo database. It contains object definitions that cannot be compared using Application Designer. The UPGNONCOMP project for your upgrade may contain some or all objects of the following object types: trees, access groups, roles, dimensions, cube definitions, and cube instance definitions. These object definitions are required for your upgraded database to function correctly. You need to review this project to see whether you customized any of the objects. You then need to reapply those customizations later in the upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All



## CHAPTER 4

# Apply Application Changes

This chapter discusses:

- Understanding Application Changes
- Customizing the New Release
- Running New Release Upgrade Copy
- Updating Database Overrides
- Backing Up After the Upgrade Copy
- Modifying the Database Structure
- Loading Data for Data Conversion
- Restoring New Release Demo
- Applying Updates Before Data Conversion
- Configuring Scheduler and Server
- Backing Up Before Data Conversion
- Running Data Conversion
- Backing Up After Data Conversion
- Finalizing the Database Structure
- Loading Data to Complete System Setup
- Loading Stored Statements
- Running Final Update Statistics
- Completing Application Processes
- Updating Language Data
- Updating Object Version Numbers
- Running the Final Audit Reports

---

## Understanding Application Changes

Earlier in the upgrade, you made various application changes. Now it is time to apply these application changes to your Copy of Production database.

## Task 4-1: Customizing the New Release

This section discusses:

- Understanding New Release Customizations
- Copying the UPGCUST Project
- Reviewing Copy Results

### Understanding New Release Customizations

In this task, the UPGCUST project is copied from the Copy of Production database to the Demo database.

**Note.** If you are running Sybase, check the configuration parameter for open objects. If this parameter is set too low, you may encounter the following error: `ct_connect(): network packet layer: internal net library error during the compare or copy process`. If you encounter this error, you need to increase the value of this parameter accordingly.

### Task 4-1-1: Copying the UPGCUST Project

This step copies your customized PeopleTools and application objects from the Copy of Production database to your Demo database.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

### Task 4-1-2: Reviewing Copy Results

Review the results of the project copies that were performed in this task. For each of the projects copied, review the copy logs for any errors. Also, verify in Application Designer that each of the projects copied shows the Done options are checked for those objects you expected to be copied.

There are many different errors you can find in the copy logs, depending on which objects you chose to copy or not copy. For example, if you chose not to copy a record definition, but neglected to clear the PeopleCode Upgrade check box for that record, you will receive errors when trying to copy the PeopleCode. Application Designer maintains PeopleTools integrity during the copy and will not copy PeopleCode for records that do not exist.

Review any errors you receive during the copy process and determine whether they are acceptable cases or unacceptable errors that need correction. In the example above, either the PeopleCode error is acceptable because you do not intend to copy the record definition, or the error is unacceptable and you should copy the record and then copy the PeopleCode for that record again.

You may get messages similar to “Warning: FIELDNAME is a key field and has been appended to the end of the RECORDNAME record.” This is an acceptable message and you can ignore it.

The following error occurs when copying a Portal Registry Structure that has a different PORTAL\_OBJNAME but the same PORTAL\_URLTEXT as an existing registry object.

Duplicate Key. Portal: *portalname*, Obj name: *objectname*, CP: *nodename*, URL (1st 50⇒  
char): *URL*

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

---

## Task 4-2: Running New Release Upgrade Copy

This section discusses:

- Exporting Selected PeopleTools Tables
- Importing Selected PeopleTools Tables
- Copying the UPGCUSTIB Project
- Swapping PeopleTools Tables
- Creating the UPGIBCOPY Project
- Copying the UPGIBCOPY Project
- Updating Target Values
- Copying the UPGNONCOMP Project
- Reviewing Project Copy Results
- Exporting New Release Objects
- Importing New Release Objects
- Resetting Object Version Numbers
- Backing Up New Release Demo Again

### Task 4-2-1: Exporting Selected PeopleTools Tables

Depending on your upgrade path you will need to export one or more PeopleTools tables to preserve values on your Copy of Production database. This step exports PeopleTools tables in the Copy of Production before the upgrade copy has occurred.

The script for your upgrade path is:

```
DLUPX96E.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-2-2: Importing Selected PeopleTools Tables

Depending on your upgrade path you will need to import one or more PeopleTools tables to preserve values on your Copy of Production database. This step imports PeopleTools tables into the Demo database before the upgrade copy occurs.

The script for your upgrade path is:

```
DLUPX96I.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-3: Copying the UPGCUSTIB Project

This step copies your customized Integration Broker objects from the Copy of Production database to your Demo database.

See Customizing the New Release, Reviewing Copy Results.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-4: Swapping PeopleTools Tables

This step swaps the base language for tables that contain PeopleTools Managed Object data and related language data on your Demo database. This is in preparation for the step, “Exporting New Release Objects.” This script should only be run if your Copy of Production has a base language other than English. The script name for your upgrade path is:

```
PT_RELEASE_SWAP.DMS
```

If you would like to automate this step, follow the procedure below.

To make this step automated:

1. Select the step Swapping PeopleTools Tables in Change Assistant.
2. Open the Step Properties dialog box.
3. Change the Type from *ManualStop* to *DataMoverUser*.

4. Click OK.
5. In your upgrade job, mark the step as Run.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	Non-English Base Language

## Task 4-2-5: Creating the UPGIBCOPY Project

In this step, you create the UPGIBCOPY project and use it to upgrade your Integration Broker objects to the new release.

To create the UPGIBCOPY project:

1. Launch PeopleTools and sign on to the New Release Demo database.
2. From Application Designer, select File, New...
3. Select Project, and then click OK.
4. Select Insert, Definitions into Project...
5. Select *Messages* from the Object Type drop-down list box.
6. Click Insert, and then click Select All.
7. Click Insert.
8. Repeat steps 5, 6, and 7 for the following object types:
  - Services
  - Service Operations
  - Service Operation Handlers
  - Service Operating Versions
  - Service Operation Routings
  - IB Queues
9. Click Insert, and then click Close.
10. Select File, Save All.
11. Enter the project name UPGIBCOPY.

---

**Warning!** You must name the project UPGIBCOPY or the next step will fail.

---

12. Click OK.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-6: Copying the UPGIBCOPY Project

In this step, copy all objects in the project, UPGIBCOPY. This project consists of Oracle-delivered Integration Broker objects as well as any customizations you may have selected to copy to the New Release Demo in a previous step.

See Copying the UPGCUSTIB Project.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-2-7: Updating Target Values

This step updates the Message Node table on the Demo database to keep the assignment of the Local Node defined in the Copy of Production. The update uses the copy of the Message Node table taken earlier in the upgrade.

The script for your upgrade path is:

```
DLUPX97.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-8: Copying the UPGNONCOMP Project

In this step, copy the non-compare project, UPGNONCOMP. This project consists of object types you cannot compare and object types not included in your compare project. In a previous step, you reviewed this Oracle-delivered project and modified the Upgrade check box for any objects you did not want to copy.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-2-9: Reviewing Project Copy Results

Review the results of the UPGCUSTIB, UPGIBCOPY, and UPGNONCOMP project copy steps that were performed earlier in this task. Review each copy log for any errors and verify in Application Designer that the Done options are checked for the objects in each of the projects. To verify Done options for UPGIBCOPY and UPGNONCOMP, you would log in to the Demo database, whereas to verify Done options for UPGCUSTIB, you would log in to your Copy of Production database.

There are many different errors you can find in the copy logs, depending on which objects you chose to copy or not copy. Review any errors you received during the copy process to determine whether they are acceptable cases or unacceptable errors that need corrective action.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-10: Exporting New Release Objects

This step exports the new release objects and your customizations that you copied to the Demo database in an earlier step, to a file.

The script name for your upgrade path is:

```
PT_RELEASE_EXPORT.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-2-11: Importing New Release Objects

This step imports the new release objects and your customizations into your Copy of Production database.

The script name for your upgrade path is:

```
PT_RELEASE_IMPORT.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-2-12: Resetting Object Version Numbers

In this step, you run the VERSION Application Engine program. This ensures that all of your version numbers are correct and, if not, resets them to 1.

---

**Note.** You will rerun the VERSION Application Engine program later in the upgrade. If you want to preserve the log files generated by Change Assistant from this run, you will need to manually rename the files after completing this step.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-2-13: Backing Up New Release Demo Again

Back up your New Release Demo database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remainder of the tasks in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

---

## Task 4-3: Updating Database Overrides

This section discusses:

- Understanding Database Overrides
- Set Index Parameters After Copy
- Set Tablespace Names After Copy
- Set Record Parameters After Copy
- Create New Tablespaces

### Understanding Database Overrides

In this task, you update PeopleTools tables with DDL information from your physical database DDL. You may have overwritten information about where tables exist in your database during the copy project steps of this upgrade. The following steps synchronize your PeopleTools table definitions with your database again.

#### Task 4-3-1: Set Index Parameters After Copy

This step updates index overrides stored in the PSIDXDDLARM table. The values stored in the PARMVALUE field are updated with current values found in the system catalog. The name of the process is:

```
SETINDEX.SQR
```



## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle DB2 z/OS	All

## Task 4-3-2: Set Tablespace Names After Copy

This step updates tablespace names stored in the PSRECTBLSPC table. In addition, the values stored in the DDLSPACENAME field are updated with current values found in the system catalog. If you modified tablespace names from the Oracle-delivered names, this process makes those same changes in the PeopleSoft system record definition. It also corrects any tablespace names that were reset with values from the Demo database during the copy project step. The process then lists any tablespaces defined in the PeopleTools tables that are not currently on your database. Use this report to create new tablespaces later in this task. The name of the process is:

SETSPACE.SQR

---

**Note.** This step updates both the database and tablespace names in the PSRECTBLSPC table for DB2 z/OS sites. The report produced by this process lists database/tablespace combinations that were not defined in the DB2 system catalog. The report may show your Demo database and tablespace names instead of your Copy of Production database and tablespace names. You will correct this situation when you create new tablespaces.

---

See Create New Tablespaces.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 4-3-3: Set Record Parameters After Copy

This step updates table overrides stored in the PSRECDDLARM table. The values stored in the PARMVALUE field are updated with the current values found in the system catalog. The name of the process is:

SETTABLE.SQR

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle	All

## Task 4-3-4: Create New Tablespaces

This section discusses:

- Prerequisites
- Creating Oracle Delivered Tablespaces
- Creating Custom Tablespaces

### Prerequisites

Before you perform this step, you must make sure your database administrator has created all new tablespaces that will be used in new tables.

---

**Note.** DB2 z/OS sites need to create databases as well as tablespaces at this time.

---

### Creating Oracle Delivered Tablespaces

If you use Oracle-delivered tablespace names, be aware that there may be new ones in this release. The report that you produced when you set tablespace names after copying provides a list of tablespaces that are missing from your database.

See Set Tablespace Names After Copy.

You need to create all the tablespaces on the report listed as missing on the database. Once you create all the tablespaces, you can rerun the SETSPACE.SQR; the report should show that no additional modifications are needed.

---

**Note.** For DBX sites, create all the tablespaces on the report listed as missing on the database in addition to the corresponding index (IDX) tablespace.

---

Oracle delivered a shell SQL script containing the DDL commands to create all the Oracle-delivered tablespaces. Edit the script to create just the new tablespaces and to set up the script for your environment.

---

**Note.** Oracle reassigned some tables to larger tablespaces because they now require a 32-KB buffer pool. You must manually edit the Create Table statements in the upgrade scripts to replace the tablespace names with an appropriate tablespace name in your implementation that utilizes a 32-KB buffer pool.

---

The script name is:

```
HCDDL.SQL (HCDDL.U.SQL for DB2 z/OS Unicode)
```

---

**Important!** For DB2 UNIX/NT sites, the script name is HCDDL.DMS.SQL for ANSI, and HCDDL.DMSU.SQL for Unicode.

---

DB2 z/OS sites must also consider how database names are assigned. After the upgrade/copy is completed, some of the PeopleTools metadata tables in your Copy of Production database will contain the database values from the Demo database. Review the SETSPACE SQR report for those tables that are reported as not defined in the database. If the report shows your Demo database names instead of your Copy of Production database names you can reset them with the following SQL:

```
UPDATE PSRECTBLSPC SET DBNAME = 'Copy of Production dbname'
WHERE DBNAME = 'Demo dbname'
```

## Creating Custom Tablespaces

If you will use custom tablespaces, create those tablespaces now. Choose one of the following two methods to get the information into PeopleTools:

- Update PeopleTools for each record you will put into a custom tablespace. You can do this directly through Application Designer, or you can update PSRECTBLSPC directly by using the appropriate SQL for your site, as follows:

DB2 z/OS sites:

```
UPDATE PSRECTBLSPC
SET DBNAME = 'new dbname', DDLSPACENAME = 'new tablespacename'
WHERE DBNAME = 'current dbname'
AND DDLSPACENAME = 'current tablespacename';
```

All other sites:

```
UPDATE PSRECTBLSPC
SET DDLSPACENAME = 'new tablespacename'
WHERE DDLSPACENAME = 'current tablespacename';
```

To update each table individually, add the following clause to the predicate of the above statement, making sure you use the record name in this clause:

```
AND RECNAME = record name
```

The SETSPACE report contains the table name. The record name will not have the “PS\_” prefix.

You can double-check that you created all tablespaces by rerunning the SETSPACE.SQR report. If you created all tablespaces for records defined in PeopleTools, the report will be empty.

- When you edit the Create and Alter scripts, you can change the SQL to create the tables in the correct tablespaces. Later in this task you will set tablespace names, which will update PeopleTools with the correct tablespaces or database/tablespace in DB2 z/OS. The report should be empty at that time.

---

**Note.** For DB2 z/OS sites, the SETSPACE report may list some database/tablespace combinations as “Table Undefined - DB/TS OK” when in fact the database name is one that was defined for your Demo database. This occurs if your Demo and Copy of Production databases are in the same DB2 subsystem. The SETSPACE.SQR detected that the database/tablespace combinations do exist in the subsystem and are therefore valid. Make sure that you update these database/tablespace names to match those that exist in your Copy of Production, using the instructions above.

---



---

**Note.** During the Move to Production pass, you will create these tablespaces when you populate tablespace data. You can reuse this script, or you can create a new script for your production environment. To reuse the script you have created for this task, save it and copy it into the PS\_HOME\SCRIPTS directory that you use during the Move to Production pass.

---

See the Enterprise PeopleTools installation guide for DB2 UDB for z/OS for your new release, “Creating a Database,” Correcting Invalid Database/Tablespace Combinations.

See Modifying the Database Structure, Editing the Create and Alter Scripts.

See Modifying the Database Structure, Setting Tablespace Names.

See “Apply PeopleTools Changes,” Populating Tablespace Data.

See “Apply Changes to Production Database,” Performing the Move to Production.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle Informix DB2 z/OS DB2 UNIX/NT	All

---

## Task 4-4: Backing Up After the Upgrade Copy

Back up your database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remaining tasks in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 4-5: Modifying the Database Structure

This section discusses:

- Understanding Modifying the Database Structure
- Backing Up for DB2
- Building the Upgrade Tables Script
- ReCreating Upgrade Tables
- Building Renamed Tables Script
- Running the Renamed Tables Script
- Creating the ALLTEMPTABS Project

- Building the Create Temp Tables Script
- Creating the ALLTABS Project
- Building the Create and Alter Scripts
- Editing the Create and Alter Scripts
- Creating Temporary Tables
- Creating Tables
- Altering Tables
- Creating Indexes
- ReCreating Triggers
- Reviewing the Create Indexes Log
- Dropping Indexes for Data Conversion
- Creating Indexes for Data Conversion
- Setting Index Parameters
- Setting Temp Table Tablespace Names
- Setting Tablespace Names
- Setting Record Parameters
- Generating a DB2 UNIX RUNSTATS Script
- Updating Statistics for DB2 UNIX
- Updating Statistics for DB2 zOS
- Updating Statistics for Informix
- Updating Statistics for Oracle

## Understanding Modifying the Database Structure

In this task you create and run various scripts and processes that will modify your database structure, including creating new tables and indexes, altering tables that have changed, and re-creating modified indexes.

---

**Note.** In the Change Assistant job, some of the steps may complete without error, but display a Warning icon indicating that warning messages exist in the log file.

---

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release, “Error Handling.”

### Task 4-5-1: Backing Up for DB2

If you are using the DB2 z/OS platform, back up your database now. This enables you to restart your upgrade from this point if you should experience any database integrity problems during the remaining tasks in the upgrade process.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	DB2 z/OS	All

## Task 4-5-2: Building the Upgrade Tables Script

This step generates the SQL script to drop and re-create all the tables in the project named UPGCONVERT. These tables will be used during data conversion by Application Engine programs. They can be safely dropped at this time because they do not contain application data required by your PeopleSoft system.

The script name for your upgrade path is:

UPGCONVERT\_CRTTBL.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-3: ReCreating Upgrade Tables

This step runs the SQL script you generated to re-create all the tables in the project named UPGCONVERT.

The script name for your upgrade path is:

UPGCONVERT\_CRTTBL.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-4: Building Renamed Tables Script

In this step, you will create the SQL script to alter the tables that were renamed with Data Mover rename scripts in an earlier task. Data Mover only renamed the records in Application Designer. This step will build the SQL to perform the corresponding rename at the database level.

The generated script name is:

UPGRNCS80.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-5: Running the Renamed Tables Script

In this step, you will run the script to alter the tables that were renamed in the Data Mover rename scripts in an earlier task.

The script name for your upgrade path is:

UPGRNCS80.SQL

When the Change Assistant runs the Create Indexes script to create indexes, it will not stop when it encounters errors. You may view the log file, and will see some indexes cannot be created due to unique index constraints. The data causing those indexes to fail will be updated during the task, “Running Data Conversion.” The indexes will then create successfully during the task, “Finalizing the Database Structure.” The log for your upgrade path is:

UPGRNHC80.LOG

See Running Data Conversion.

See Finalizing the Database Structure.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-6: Creating the ALLTEMPTABS Project

This step creates a project named ALLTEMPTABS and inserts all records of the type *Temporary Table*.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-7: Building the Create Temp Tables Script

This step generates the SQL script to drop and re-create all the records of type Temporary Table in the database. Processes use the Temporary Tables dynamically in your system. They can be safely dropped at this time because they do not contain transaction data required by your PeopleSoft system.

The script name for your upgrade path is:

ALLTEMPTABS\_CRTTBL.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-8: Creating the ALLTABS Project

This step creates a project named ALLTABS and inserts all records of the type *Table*.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-9: Building the Create and Alter Scripts

This step generates the SQL script to create all new records of the type *Table*. The script name is:

```
ALLTABS_CRTTBL.SQL
```

This step generates the SQL script to alter all existing records of the type *Table*. This script is referred to as Alter Without Deletes. The tables are altered to add new columns, rename existing columns and change columns that have modified properties, such as length. Columns that will eventually be deleted will still exist on the tables after this script is executed. The script name is:

```
ALLTABS_ALTTBL.SQL
```

This step also generates the SQL script to create new indexes and to re-create modified indexes as needed for the tables in the first two scripts. The script name is:

```
ALLTABS_CRTIDX.SQL
```

---

**Note.** This step also creates the script ALLTABS\_CRTTRG.SQL, which re-creates all database triggers. You do not need to run this script, because all database triggers will be created in the “Finalizing the Database Structure” task.

---



---

**Note.** For DB2 z/OS sites, if this step takes an exceptionally long time, performing a RUNSTATS on the system catalog tablespace SYSDBASE may improve performance.

---

See Finalizing the Database Structure.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All



## Task 4-5-10: Editing the Create and Alter Scripts

In this step, you will edit the SQL create and alter scripts for tablespace names and sizing. The script names for your upgrade path are:

```
ALLTEMPTABS_CRTTBL.SQL
ALLTABS_CRTTBL.SQL
ALLTABS_ALTTBL.SQL
ALLTABS_CRTIDX.SQL
```

If you are not using the PeopleSoft tablespace names, you will need to review and modify the scripts above. When the new record was copied to the Copy of Production database, the PeopleSoft default tablespace name was copied as well. When you performed the step, “Create New Tablespaces,” you were given the option to correct the tablespace names online or to wait and edit the scripts. After you have completed running these scripts you will run the programs that synchronize the PeopleTools definitions with the database catalog again. Therefore, any changes you make to the scripts now will be reflected in the PeopleTools definition. Have your database administrator review these scripts and modify the tablespace names appropriately.

Many of the new tables and indexes will be populated during the upgrade. If they are not sized appropriately for your database, the conversion programs will stop with errors. After the upgrade is complete, you may want your database administrator to review and make adjustments to the amount of free space left in some of the tables or tablespaces.

PeopleSoft has provided a list of these new tables with information about how to size them relative to the size of your existing data. This information is meant to give you a rough estimate on an initial size, so you can execute the conversion programs without errors.

See Appendix: “Sizing Tables for the Upgrade.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS DB2 UNIX/NT Oracle Informix	All

## Task 4-5-11: Creating Temporary Tables

This step runs the SQL script you generated to create all the records of the type *Temporary Table*. The script name for your upgrade path is:

```
ALLTEMPTABS_CRTTBL.SQL
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-12: Creating Tables

This step runs the SQL script you generated to create all the records of the type *Table*. This step creates new table structures in your database. The script name for your upgrade path is:

ALLTABS\_CRTTBL.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-13: Altering Tables

This step runs the SQL script you generated to alter the existing records of type *Table*. This step alters existing PeopleSoft table structures to comply with your new PeopleSoft release.

The script name for your upgrade path is:

ALLTABS\_ALTTBL.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-14: Creating Indexes

This step runs the SQL script you generated to create indexes on records of the type *Table*. This step creates or modifies indexes as required.

The script name for your upgrade path is:

ALLTABS\_CRTIDX.SQL

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-15: ReCreating Triggers

This step executes the script CREATETRGR.DMS, which will re-create all PeopleSoft triggers in the database. The triggers on PeopleSoft tables were invalidated when the tables were altered and need to be re-created.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 4-5-16: Reviewing the Create Indexes Log

When Change Assistant runs the create indexes script to create indexes, it will not stop when it encounters errors. When you view the log file, you will see that some indexes cannot be created due to unique index constraints. The data causing those indexes to fail will be updated during the task, “Running Data Conversion.” The indexes will then create successfully during the task, “Finalizing the Database Structure.”

Review the errors in the log file. Unique constraint errors are acceptable. If you see any other types of index creation errors, such as space problems, you must correct them before you continue with the upgrade. If you do not correct the errors, it may degrade your performance during data conversion.

The log file name for your upgrade path is:

ALLTABS\_CRTIDX.LOG

See Running Data Conversion.

See Finalizing the Database Structure.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-17: Dropping Indexes for Data Conversion

Drop the following indexes using the SQL tool of your choice. These tables are changing key structure and data conversion cannot update the new columns if these old indexes exist. Some of these indexes may not exist because they did not create in the earlier Create Indexes step. This is acceptable.

Table Name	Index Name
PS_LIMIT_TBL	PS_LIMIT_TBL
PS_LIMIT_INCLD_TBL	PS_LIMIT_INCLD_TBL
PS_LIMIT_EXCLD_TBL	PS_LIMIT_EXCLD_TBL
PS_LIMIT_IMPIN_TBL	PS_LIMIT_IMPIN_TBL
PS_STEPINC_TBL_JPN	PS_STEPINC_TBL_JPN

Table Name	Index Name
PS_SAL_AMT_TBL_JPN	PS_SAL_AMT_TBL_JPN
PS_CONTR_REPS_BEL	PS_CONTR_REPS_BEL
PS_CONTR_ENDS_BEL	PS_CONTR_ENDS_BEL
PS_ETHNICITY_DTL	PS_ETHNICITY_DTL
PS_SAD_LAW_CD_TST	PS_SAD_LAW_CD_TST
PS_SCH_ADHOC_DTL	PS_SCH_ADHOC_DTL
PS_HRS_APP_CMP	PS_HRS_APP_CMP

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-18: Creating Indexes for Data Conversion

During this step, you will create indexes to improve performance during data conversion. These indexes are only used for the purposes of data conversion. They are not required for regular operation of the products, therefore they were not created in previous steps nor are they delivered with the record definitions on the Demo database. In this step you are asked to manually create those indexes that apply to you (not using Application Designer).

It is important to follow the following naming convention so the Alter with Deletes script you run in the task “Finalizing Database Structure” will drop these indexes automatically. Use PS[X]Record Name where X is any letter A-Z. For example, if you have to create an index for table PS\_COUNTRY\_TBL a proper index name would be PSUCOUNTRY\_TBL. If an additional index is required for that same table, you can then name it (for example) PSYCOUNTRY\_TBL. However, it is important to ensure that the new name of the index you are creating is not defined in the database already.

Not all listed indexes are required for all customers. Reading the comments with each listed index will help you determine if the index is applicable to your specific upgrade. Work with your database administrator to create a script that will build the indexes in your Copy of Production. You can then modify the Change Assistant Template to automate this step (for the initial pass and all future passes too). To do that, go to the Upgrade Template view, then right-click on the step and edit the step properties. Add your script name, without the file extension, in the Script/Procedure field and change the Type to SQL Script.

After reading the comments to determine whether the index is applicable to your situation, create or modify the following indexes in your Copy of Production database:

Table	With Columns	Comments
PS_APPLICATN_ROUTE	APPLID APP_DT JOB_REQ_NBR ROUTE_DT	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRTE1.
PS_APP_PRIWRK_EXPR	APPLID APP_DT	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HCERW202.
PS_HRS_APPLICANT	APPLID	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HROIALSR.STEP34.
PS_HRS_RCMNT_TAO	HRS_PERSON_ID HRS_JOB_OPENING_ID HRS_PROFILE_SEQ	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRCM1.UPDIDs.
PS_HRS_RCMS_TAO	HRS_PERSON_ID HRS_RCMNT_ID STATUS_DT SEQ_NUM	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRCM6.
PS_HRS_RTED_TAO	HRS_PERSON_ID HRS_RCMNT_ID HRS_RTE_ID HRS_RTE_SEQ STATUS_CODE STATUS_DT SEQ_NUM	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRTE4.
PS_HRS_RTES_TAO	HRS_PERSON_ID HRS_RCMNT_ID HRS_RTE_ID STATUS_CODE STATUS_DT SEQ_NUM	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRTE3.
PS_HRS_RTE_TAO	HRS_PERSON_ID HRS_RCMNT_ID	This index should be non-unique. It will significantly improve performance for section UPG_ER88.HRATRTE1.

Table	With Columns	Comments
PS_JOB	POSITION_NBR EFFDT FLSA_STATUS	This index should be non-unique. It will significantly improve performance for section UPG_HC88.HCPYW10K.
PS_JOB	ACTION EMPLID EMPL_RCD EFFDT EFFSEQ	This index should be non-unique. It will significantly improve performance for section UPG_HC88. HCHCW12.
PS_TL_MTCHD_2	PROCESS_INSTANCE EMPLID EMPL_RCD PUNCH_BEGIN TIMEZONE DUR	This index should be non-unique. It will significantly improve performance for step UPG_TL88.HCTLWO8.Step09 (SQL).
PSTZOFFSET	TIMEZONE STARTDATETIME ENDDATETIME BASEOFFSET	This index should be non-unique. It will significantly improve performance for step UPG_TL88.HCTLWO8.Step09 (SQL).
PS_UPG_GRNOPR_TMP	STATE_GARN_LAW GARN_RULE_ID EFFDT GARN_STEP	This index should be non-unique. It will improve the performance of UPG_PY89.HCPYT05 for North American Payroll customers.
PS_UPG_GRNRL_TMP	STATE_GARN_LAW GARN_RULE_ID EFFDT	This index should be non-unique. It will improve the performance of UPG_PY89.HCPYT05 for North American Payroll customers.
PS_UPG_GRNXMP_TMP	STATE_GARN_LAW GARN_RULE_ID EFFDT GARN_EX_CALC	This index should be non-unique. It will improve the performance of UPG_PY89.HCPYT05 for North American Payroll customers.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-5-19: Setting Index Parameters

This step updates index overrides stored in the PSIDXDDLARM table. The values stored in the PARMVALUE field are updated with current values found in the system catalog. The name of the process is:

SETINDEX.SQR

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle DB2 z/OS	All

## Task 4-5-20: Setting Temp Table Tablespace Names

This step populates the PeopleTools table PSRECTBLSPC with the table name, database name, and tablespace name information for the temporary table instances created on the database in a previous step. This information will be required by processes that perform in-stream RUNSTATS (%UpdateStats) on the temporary table instances. The name of the process is:

SETTMPIN.SQR

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 4-5-21: Setting Tablespace Names

This step populates all tablespace information in the PSRECTBLSPC table. The values stored in the DDLSPACENAM field are updated with current values found in the system catalog. If you modified tablespace names when you edited the SQL script that created your new tables from the Oracle-delivered names, this will make those same changes in the PeopleSoft record definition. The name of the process is:

SETSPACE.SQR

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle Informix DB2 UNIX/NT DB2 z/OS	All

## Task 4-5-22: Setting Record Parameters

This step updates table overrides stored in the PSRECDDLPRM table. The values stored in the PARMVALUE field are updated with the current values found in the system catalog. The name of the process is:

```
SETTABLE.SQR
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

## Task 4-5-23: Generating a DB2 UNIX RUNSTATS Script

This step executes the RUNSTATS.SQR that creates the RUNSTATS.SQL to update the statistics on DB2 UNIX/NT.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

## Task 4-5-24: Updating Statistics for DB2 UNIX

Earlier in the upgrade process, you updated your statistics. Now that you have copied your new objects and created new indexes, update your statistics again. Run the RUNSTATS.SQL script created in the previous step to improve performance of your data conversions and generation of the Alter with Delete script.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All



## Task 4-5-25: Updating Statistics for DB2 zOS

Earlier in the upgrade process, you updated your statistics. Now that you have copied your new objects and created new indexes, update your statistics again to improve performance of your data conversions and generation of the Alter with Delete script. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

## Task 4-5-26: Updating Statistics for Informix

Earlier in the upgrade process, you updated your statistics. Now that you have copied your new objects and created new indexes, update your statistics again to improve performance of your data conversions and generation of the Alter with Delete script.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All

## Task 4-5-27: Updating Statistics for Oracle

Earlier in the upgrade process, you updated your statistics. Now that you have copied your new objects and created new indexes, update your statistics again to improve performance of your data conversions and generation of the Alter with Delete script. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

---

## Task 4-6: Loading Data for Data Conversion

This section discusses:

- Swapping Languages on System Data
- Exporting Application Messages
- Importing Application Messages
- Exporting Record Groups

- Importing Record Groups
- Exporting the System Setup Data
- Importing the System Setup Data
- Exporting the PW Pagelet Data
- Importing the PW Pagelet Data
- Exporting the Pagelet Wizard Data
- Importing the Pagelet Wizard Data
- Exporting Upgrade Defaults
- Importing Upgrade Defaults
- Exporting Delivered Equations
- Importing Delivered Equations
- Exporting Application Conversion Data
- Importing Application Conversion Data
- Exporting Data Conversion Driver Data
- Importing Data Conversion Driver Data
- Populating Control Tables for Data Conversion

## Task 4-6-1: Swapping Languages on System Data

This script swaps the base language for tables that contain system data on your Demo database and have related language data, in preparation for the system data exports in the next step. This script should be run only if your Copy of Production has a base language other than English. The script name for your upgrade path is:

`DLHCLASWAP.DMS`

If you want to make this step automated, follow the steps below.

To make this step automated:

1. Select the step Swapping Languages on System Data in Change Assistant.
2. Open the Step Properties dialog box.
3. Change the Type from *ManualStop* to *DataMoverUser*.
4. Click OK.
5. In your upgrade job, mark the step as Run.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	Non-English Base Language

## Task 4-6-2: Exporting Application Messages

This step exports Application Messages data from the Demo database. The script name for your upgrade path is:

DLUPX01E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-6-3: Importing Application Messages

This step imports Application Message data into your Copy of Production database. Message Sets 0-999 are overlaid during the PeopleTools Upgrade. Application Message Sets 1000-19,999 are overlaid with this task. If you have added custom messages in this set range, you must add those messages again at the end of the upgrade. To prevent this from happening again in future maintenance or upgrades, add your custom messages in a set range of 20,000 or greater.

---

**Note.** If the script fails, verify that your Configuration Manager Profile output and input directories are set to the same location. If not, this could be the cause of the problem.

---

The script name for your upgrade path is:

DLUPX01I.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-6-4: Exporting Record Groups

This step exports Record Group data from the Demo database. The script name for your upgrade path is:

DLUPX02E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-6-5: Importing Record Groups

This step imports Record Group data and populates Set Control data in your Copy of Production database. The following records are related to Record Groups and Set Control data:

- REC\_GROUP\_REC
- REC\_GROUP\_TBL
- SET\_CNTRL\_TBL
- SET\_CNTRL\_GROUP
- SET\_CNTRL\_REC
- SETID\_TBL

The import script deletes from, and then reloads, the Record Group tables, REC\_GROUP\_REC and REC\_GROUP\_TBL. These are the tables that are modified when you use PeopleTools, Utilities, Administration, Record Group. The script then rebuilds the related setID tables, PS\_SET\_CNTRL\_GROUP and PS\_SET\_CNTRL\_REC. The PS\_SET\_CNTRL\_TBL and PS\_SETID\_TBL tables contain the setIDs you use in your system; this script does not update PS\_SET\_CNTRL\_TBL. However, it does check for orphan setID references in PS\_SET\_CNTRL\_REC and adds the missing setIDs to PS\_SETID\_TBL.

If you have moved an Oracle-delivered record into a custom added record group, and deleted the record from the Oracle-delivered record group, this script will put the record back into the Oracle-delivered record group and remove it from the custom added record group.

If you have created a new record group, it will be deleted in this step if all of its records are assigned to Oracle-delivered record groups in the new release. To continue using your custom record group, you will need to re-create it in the Reapply Customizations task.

This script creates an output file and uses it to create a temporary table. To run successfully, the Configuration Manager input and output Data Mover directories should be the same.

---

**Note.** If the script fails, verify that your Configuration Manager Profile output and input directories are set to the same location. If not, this could be the cause of the problem.

---

The script name for your upgrade path is:

DLUPX02I.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-6-6: Exporting the System Setup Data

This script exports the contents of the Message, Strings, Stored Statements, Record Group, data conversion driver, EDI, and Mass Change tables from the Copy of Production database during your Move to Production passes. During the initial pass, you ran other scripts to load this data and in some cases had to reapply customizations. This script exports the entire contents of these tables, including customizations, so that you will not need to reapply them after the Move to Production. The script name for your upgrade path is:

MVAPPEXP.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	All	All	All

## Task 4-6-7: Importing the System Setup Data

This script imports the data exported in the previous step into your New Copy of Production database during your Move to Production passes. This script replaces many scripts that you ran in the initial pass. It will move all data in these tables so that any customizations you have added to these tables during your initial pass will be moved to your New Copy of Production database. Also, it will rebuild the Set Control tables using the Record Groups from the Copy of Production database and your current Set Control values on the New Copy of Production database. The script name for your upgrade path is:

MVAPPIMP.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 4-6-8: Exporting the PW Pagelet Data

This script exports the application-specific Pagelet Wizard pagelet definition, header, footer, and category tables from the Demo database in the initial pass. The script name is:

DLUPX14E.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-6-9: Importing the PW Pagelet Data

This script imports the application-specific data for the Pagelet Wizard pagelet definition, header, footer, and category tables into your Copy of Production database during the initial pass. This data is needed for the data conversion. The script name is:

DLUPX14I.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-6-10: Exporting the Pagelet Wizard Data

This script exports the contents of the Pagelet Wizard tables from the Copy of Production database during your Move to Production passes. During the initial pass, you ran programs and scripts to load this data and in some cases had to make changes. This script exports the entire contents of these tables, including changes, so that you will not need to reapply them after the Move to Production. This data is needed for the data conversion. The script name is:

MVUPX16E.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	All	All	All

## Task 4-6-11: Importing the Pagelet Wizard Data

This script imports the Pagelet Wizard tables from the Copy of Production database into the New Copy of Production during your Move to Production passes. This script replaces processes that you ran in the initial pass. It will move all data in the affected tables so that any changes you have made during your initial pass will be moved to your New Copy of Production database. This data is needed for the data conversion. The script name is:

MVUPX16I.DMS

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 4-6-12: Exporting Upgrade Defaults

This script exports the upgrade default data values and mapping during your Move to Production passes. This is the data that you set up during the chapter “Prepare Your Database” of your initial upgrade pass. You will load this information into your New Copy of Production later in the Move to Production upgrade pass. The script name for your upgrade path is:

MVCS80EXP.DMS

See “Prepare Your Database.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	All	All	All

## Task 4-6-13: Importing Upgrade Defaults

This script imports the upgrade default data values and mapping that you set up during the chapter “Prepare Your Database,” of your initial upgrade pass. The script name for your upgrade path is:

MVCS80IMP.DMS

See “Prepare Your Database.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 4-6-14: Exporting Delivered Equations

This script exports the equations delivered in the new PeopleSoft release to copy them to your Copy of Production database. During data conversion your equations will be converted to the new structure but the equations delivered in the previous release will not be upgraded. The script name for your upgrade path is:

DLCSFAG02E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 4-6-15: Importing Delivered Equations

This script imports the equations delivered in the new PeopleSoft release Demo database. During data conversion your equations will be converted to the new structure but the equations delivered in the previous release will not be upgraded. The script name for your upgrade path is:

DLCSFAG02I.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-6-16: Exporting Application Conversion Data

In this step, you will export data, from the Demo database, required for data conversion. The script name for your upgrade path is as follows:

```
DLHCUPT01E.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 4-6-17: Importing Application Conversion Data

In this step, you will import data into your Copy of Production database for use during data conversion. The script name for your upgrade path is as follows:

```
DLHCUPT01I.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-6-18: Exporting Data Conversion Driver Data

This step exports data conversion Application Engine driver data from the Demo database. The script name for your upgrade path is:

```
DLUPX03E.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-6-19: Importing Data Conversion Driver Data

This step imports data conversion Application Engine driver data into your Copy of Production database.

The script name for your upgrade path is:

```
DLUPX03I.DMS
```



**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

**Task 4-6-20: Populating Control Tables for Data Conversion**

This step runs the UPG\_DATACONV Application Engine program for Group 99 that will populate control tables required for data conversion by multiple data conversion groups, allowing concurrent processing of data conversion groups.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

**Task 4-7: Restoring New Release Demo**

Restore your New Release Demo database from the backup you took in Chapter 1 earlier in the upgrade. The backup was taken before projects were copied and scripts were run against the Demo database. This is done to restore the environment to the original Demo database.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

---

**Task 4-8: Applying Updates Before Data Conversion**

You should have downloaded and applied Required For Upgrade updates just after you installed your Demo database. Now is a great time to check PeopleSoft Customer Connection again for any new postings, and apply them now. Before data conversion, you should also verify that you have the most current UPGCONVERT.EXE.

See Oracle/PeopleSoft Customer Connection (Implement, Optimize + Upgrade, Upgrade Guide, Upgrade Documentation and Software, Upgrade Documentation and Scripts, Release, HRMS, PeopleSoft 9.0, <upgrade path>, Upgrade Scripts and Templates).

This is just one place that you can apply updates. There are other places in the upgrade process where applying updates may be applicable as well. How you apply the update varies depending on where you are in the upgrade.

See Appendix: “Applying Fixes Required for Upgrade.”

---

**Important!** Apply all fixes listed under the product line/release, even if you have not licensed the product the fix is listed under. There are many interdependencies between products and database objects. If you do not apply the fix, you may be introducing another error in a different area of the conversion code.

---

To apply PeopleSoft project fixes before data conversion:

1. Download Required for Upgrade Change Packages using the “Download Change Package” functionality in Change Assistant.
2. Use Change Assistant to install and apply the updates into your Demo database for this upgrade pass. Review the documentation included with each update prior to applying the update.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your current release.

3. The project is now loaded on your Demo database. You should run a project compare to make sure the objects in the fix will not overwrite any of your customizations. If you find customizations, you must decide how to deal with them before you copy the fix to your Copy of Production.
4. If you are performing a Move to Production upgrade pass, first migrate the Change Packages into the Source database for this upgrade pass. If needed, first set up Change Assistant with the environment information for your Source database. If you customized any of the objects delivered in the Change Package, you should repackage the fix to include your customizations. If you did not customize any objects delivered in the fix you may directly apply it to the Source database.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release, “Applying Updates.”

5. Migrate the Change Packages into the Target database for this upgrade pass. If needed, first set up Change Assistant with the environment information for your Target database.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 4-9: Configuring Scheduler and Server

You can manually run data conversion jobs on the server. Refer to the appendix “Improving Performance” for instructions. If you choose to run data conversion manually, configure and start your process scheduler and application servers now.

Tips for configuring and starting the application server:

- Make sure the application server domain being configured points to the Target database for this pass of the upgrade.
- Set a different JSL port for each database instance.

See the Enterprise PeopleTools installation guide for your database platform for the new release.

See Appendix: “Improving Performance.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 4-10: Backing Up Before Data Conversion

Back up your database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remainder of the tasks in the upgrade process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 4-11: Running Data Conversion

This section discusses:

- Reviewing Data Conversion Tips
- Running Data Conversion for Group 1
- Running Data Conversion for Group 2
- Running Data Conversion for Group 3
- Running Data Conversion for Group 4
- Running Data Conversion for Group 6
- Running Data Conversion for Group 7
- Running Data Conversion for Group 8
- Running Data Conversion for Group 10
- Running Data Conversion for Group 11

---

**Note.** In this task you will populate new tables and columns. Earlier, you altered tables and added all new and modified columns. You did not, however, remove obsolete columns. The following steps will move data from the obsolete columns to the new columns and tables. Later in this chapter, in the task “Finalizing the Database Structure,” you will generate and run SQL to delete those obsolete columns. The Upgrade Driver Application Engine program, UPG\_DATACONV, will run all upgrade data conversions.

---

## Task 4-11-1: Reviewing Data Conversion Tips

This section discusses:

- Upgrade Driver Program – UPG\_DATACONV
- Data Conversion Documentation
- Running Data Conversion Concurrently
- Writing Data Conversion for Your Non-Oracle Records
- Data Conversion Errors Expected During the Initial Upgrade Pass
- Restarting Data Conversion

### Upgrade Driver Program – UPG\_DATACONV

UPG\_DATACONV is an Application Engine program designed to run all upgrade data conversions. Each time the program is run during an upgrade pass, Change Assistant passes a group number parameter to the program. The program then reads the table PS\_UPG\_DATACONV, selecting all rows with that group number and ordering them by the sequence number on the row. A list of Application Engine library sections that must be run for data conversion is returned. The program then calls each section in the order of the sequence number. You can review the sections that are called by the Upgrade Driver program by accessing the Define Upgrade Drivers page on the Demo database.

### Data Conversion Documentation

Each section called by the Upgrade Driver program contains comments describing the underlying conversion. By running the UDATACNV.SQR report you can find which sections are called by the Upgrade Driver program and what they are doing.

See Appendix: “Using Data Conversion Utilities.”

### Running Data Conversion Concurrently

Each data conversion step in this task corresponds to a group number as defined in the Define Upgrade Drivers page. Each group is independent of the others unless otherwise documented, allowing the groups to be run concurrently. Oracle recommends that you run data conversion in the order it appears in your template during the initial upgrade pass to determine processing times. This is why the Change Assistant templates are delivered with the data conversion steps’ “Run Concurrently” property set to *No*. To reduce overall processing time during your Move to Production passes, you may decide to run some or all conversion steps concurrently. To run steps concurrently, reset the “Run Concurrently” property to *Yes* in your Change Assistant template.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release.

### Writing Data Conversion for Your Non-Oracle Records

The data conversion code delivered for this upgrade was written to handle only Oracle-delivered records. You may have added your own records to the system. To convert data in the underlying tables, you may need to create your own Application Engine library. The Upgrade Driver program can call an Application Engine library section that you create. To have the Upgrade Driver program call your custom section during this task, you will need to add the section on the Define Upgrade Drivers page.

See Appendix: “Using Data Conversion Utilities.”

## Data Conversion Errors Expected During the Initial Upgrade Pass

During your initial upgrade pass you can expect to have data conversion programs fail. This is because your PeopleSoft software installation is unique, which makes it difficult to write data conversions that will work for everyone all of the time. Your database may be larger than most, you may have customized Oracle-defined records, or you may not have copied all object deletions onto your Copy of Production. These differences will cause data conversion to fail. You must fix each problem on your initial Copy of Production and restart the Application Engine program. Your fixes will be automatically copied to your New Copy of Production during the Move to Production passes and data conversion will run smoothly.

If you have customized records that are delivered from Oracle, you may need to make changes to the Application Engine programs to handle these customizations. For example, here are two situations in which you may need to customize data conversion code:

- If you added fields to a Oracle-delivered record, you may need to add your additional fields to the conversion code for those records.
- If a Oracle-delivered record you customized will be deleted, you may need to add your own conversions to move the data to a new location.

Use the Find In feature of Application Designer to determine which Application Engine programs affect your customized records.

To use the Find In feature:

1. Create a project and add all Application Engine programs and related objects that have a name starting with *UPG* and save the project.
2. Select Edit, Find In.
3. Enter each customized record name in the Find What field and your project name in the Project field.
4. Click Find.

The results will appear in the output window.

Document any changes you make to data conversion programs. This way, if a new version of the program is delivered through Customer Connection, you will know exactly what changes you have made. You can then reapply the changes to the new version of the program.

If your database is large, you may have data conversion programs that fail due to running out of space as you move data from one table to another. This problem can happen on all RDBMS platforms, but is more of a problem on those platforms using tablespaces. If your data conversion terminates abnormally with a space error, examine the Application Engine SQL statements that caused the problem. Determine where the data is coming from and how much will be moved. Have your database administrator adjust the allocated space accordingly. The data conversion can then be restarted.

In the appendix, “Sizing Tables for the Upgrade,” Oracle has provided the approximate number of rows that will be loaded into some tables.

If you get a data conversion error because a field does not exist on a table, and the field is not one you have customized, check your field renames. If a field that appears on a record that is deleted in the new PeopleSoft release but was not deleted in your compare and copy, your table will be out of sync with what is expected by data conversion. If you had deleted the record, the rename would not happen on the physical table and the field would have the old name. This is what the data conversion program expects. If you did not delete the record, the field was renamed during the altering of tables and the data conversion program will terminate abnormally. Edit the Application Engine SQL to use the name, which is now on your table, and then restart the data conversion.

See Appendix: “Using the Comparison Process.”

See Appendix: “Sizing Tables for the Upgrade.”

## Restarting Data Conversion

Before restarting a data conversion step, rename the log file. Change Assistant uses the same log file name each time you start or restart an Application Engine program. This means that the restarted Application Engine program will replace the original log file if it is not renamed.

If your data conversion program fails, fix the problem on your Copy of Production and restart the program. When you set the data conversion step to Restart in your Change Assistant job, it will rerun the program using the PROCESS\_INSTANCE and RUN\_CNTL\_ID from the initial run and the conversion will restart right after the last committed SQL command. Application Engine keeps track of data committed to the database in the table PS\_AERUNCONTROL, keyed by PROCESS\_INSTANCE and RUN\_CNTL\_ID.

See Finalizing the Database Structure.

See Appendix: “Sizing Tables for the Upgrade.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-11-2: Running Data Conversion for Group 1

This step runs the UPG\_DATACONV Application Engine program for Group 1. For additional documentation for Group 1, run the UDATACNV report.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-11-3: Running Data Conversion for Group 2

This step runs the UPG\_DATACONV Application Engine program for Group 2. If you want to see additional documentation for Group 2, then run the UDATACNV report.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-11-4: Running Data Conversion for Group 3

This step runs the UPG\_DATACONV Application Engine program for Group 3. If you want to see additional documentation for Group 3, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

**Task 4-11-5: Running Data Conversion for Group 4**

This step runs the UPG\_DATACONV Application Engine program for Group 4. If you want to see additional documentation for Group 4, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

**Task 4-11-6: Running Data Conversion for Group 6**

This step runs the UPG\_DATACONV Application Engine program for Group 6. If you want to see additional documentation for Group 6, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Directory Interface	All	All

**Task 4-11-7: Running Data Conversion for Group 7**

This step runs the UPG\_DATACONV Application Engine program for Group 7. If you want to see additional documentation for Group 7, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

**Task 4-11-8: Running Data Conversion for Group 8**

This step runs the UPG\_DATACONV Application Engine program for Group 8. If you want to see additional documentation for Group 8, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

**Task 4-11-9: Running Data Conversion for Group 10**

This step runs the UPG\_DATACONV Application Engine program for Group 10. If you want to see additional documentation for Group 10, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

**Task 4-11-10: Running Data Conversion for Group 11**

This step runs the UPG\_DATACONV Application Engine program for Group 11. If you want to see additional documentation for Group 11, then run the UDATACNV report.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

**Task 4-12: Backing Up After Data Conversion**

Back up your database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remaining tasks in the upgrade process.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

**Task 4-13: Finalizing the Database Structure**

This section discusses:



- Understanding Final Database Structure
- Building the Alter with Deletes Scripts
- Altering Tables with Deletes
- Creating Indexes Again
- Creating Triggers
- Running the AE SYNCIDGEN Process
- Creating All Views

## Understanding Final Database Structure

Now that Data Conversion is completed, this task will alter the tables to remove obsolete columns, and create final indexes and views.

### Task 4-13-1: Building the Alter with Deletes Scripts

This step uses the previously created project ALLTABS and generates three SQL scripts: one that will alter tables to drop obsolete columns, one that will also create any remaining indexes that could not be created with the first alter, and one that will create triggers. The script names are:

```
ALLTABS_DEL_ALTTBL.SQL
ALLTABS_DEL_CRTIDX.SQL
ALLTABS_DEL_CRTRG.SQL
```

---

**Important!** All indexes should create when the ALLTABS\_DEL\_CRTIDX.SQL script is run. When a unique index fails to create, it is probably due to a data conversion issue. If a unique index fails to create, you must resolve the issue and not simply remove the index. To prevent this issue, you can back up tables in the ALLTABS\_DEL\_ALTTBL.SQL script that will be dropping recfields that have data. This way, if you have an issue you may have the old fields and data you need to correct it.

---



---

**Note.** For DB2 z/OS sites, if this step takes an exceptionally long time, performing a RUNSTATS on the system catalog tablespace SYSDBASE may improve performance.

---

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-13-2: Altering Tables with Deletes

This step executes the script ALLTABS\_DEL\_ALTTBL.SQL, which was generated in the previous step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-13-3: Creating Indexes Again

This step executes the script ALLTABS\_DEL\_CRTIDX.SQL, which was generated in the previous step. All indexes should create at this time.

---

**Important!** Review the log to find any unique indexes that might have failed to create. All indexes should create at this time, so those errors are not acceptable and should be corrected. When a unique index fails to create, it is probably due to a data conversion issue.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-13-4: Creating Triggers

This step executes the script ALLTABS\_DEL\_CRTTRG.SQL, which was generated in a previous step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-13-5: Running the AE SYNCIDGEN Process

This step executes the AE\_SYNCIDGEN Application Engine program. Mobile applications use Sync IDs to give each row a unique identifier. For any tables with the Sync ID column set to the default value of zero, the AE\_SYNCIDGEN program will populate the column with the next valid Sync ID value.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-13-6: Creating All Views

This step runs CREATEVW.DMS to re-create all views in the Copy of Production database. The script will try to create every view in Application Designer. If there is an error on one view, it will keep going until it gets to the end of the list.

---

**Important!** Review the log to find any views that failed to create. All views should create at this time, so those errors are not acceptable and should be corrected.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 4-14: Loading Data to Complete System Setup

This section discusses:

- Exporting Strings
- Importing Strings
- Exporting EDI Statements
- Importing EDI Statements
- Exporting Mass Change Data
- Importing Mass Change Data
- Exporting XML Service Information
- Importing XML Service Information
- Exporting Related Language System Data
- Importing Related Language System Data
- Editing the System Data Scripts
- Exporting Application System Data
- Importing Application System Data
- Exporting Data for Your Upgrade Path
- Importing Data for Your Upgrade Path
- Exporting Common Portal System Options
- Importing Common Portal System Options
- Running the DLHCCOG03E Script
- Running the DLHCCOG03I Script
- Exporting Setup Data
- Importing Setup Data
- Exporting Approval Framework Definitions

- Importing Approval Framework Definitions
- Exporting Generic Notifications
- Importing Generic Notifications

## Task 4-14-1: Exporting Strings

This script exports Strings data from the Demo database. The script name for your upgrade path is:

`DLUPX04E.DMS`

This data will be exported during Move to Production by the script `MVAPPEXP.DMS`.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-14-2: Importing Strings

This script imports Strings data into the Copy of Production database. The script name for your upgrade path is:

`DLUPX04I.DMS`

This data will be imported during Move to Production by the script `MVAPPIMP.DMS`.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-14-3: Exporting EDI Statements

This script exports EDI Statements from the Demo database. The script name for your upgrade path is:

`DLUPX05E.DMS`

This data will be exported during Move to Production by the script `MVPRDEXP.DMS`.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-14-4: Importing EDI Statements

This script imports the EDI Statements into the Copy of Production database. The script name for your upgrade path is:

DLUPX05I.DMS

This data will be imported during Move to Production by the script MVPRDIMP.DMS.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-14-5: Exporting Mass Change Data

This script exports Mass Change tables from the Demo database. The script name for your upgrade path is:

DLUPX06E.DMS

This data will be exported during Move to Production by the script MVAPPEXP.DMS.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-14-6: Importing Mass Change Data

This script imports Mass Change tables into the Copy of Production database. The script name for your upgrade path is:

DLUPX06I.DMS

This data will be imported during Move to Production by the script MVAPPIMP.DMS.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-14-7: Exporting XML Service Information

This script exports XML service data from the Demo database. The script name for your upgrade path is:

DLUPX13E.DMS

This data will be exported during Move to Production by the script MVPRDEXP.DMS.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-14-8: Importing XML Service Information

This script imports XML service data into the Copy of Production database. The script name for your upgrade path is:

```
DLUPX13I.DMS
```

This data will be imported during Move to Production by the script MVPRDIMP.DMS.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-14-9: Exporting Related Language System Data

This script exports system data from various application-related language tables in your Demo database into a Data Mover \*.DAT file. In a later step, this data will be loaded into your Copy of Production. The script name for your upgrade path is:

```
DLHCLASYSE.DMS
```

---

**Note.** During Move to Production passes you can reuse the data files that are created by this export script. Preserve this DAT file, and set the Type of Upgrade property in the Change Assistant template to Initial Upgrade for this step.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All Non-English

## Task 4-14-10: Importing Related Language System Data

This script will delete old related language system data from related language tables. The script then imports the data exported by the scripts above. The script name for your upgrade path is:

```
DLHCLASYSI.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All Non-English

## Task 4-14-11: Editing the System Data Scripts

Oracle sites will need to edit the system data scripts for date formats. Change the date format *YYYY-MM-DD* to *DD-MMM-YYYY*. For example, a date 1900-01-01 would need to be changed to 01-JAN-1900. The scripts you will need to edit are as follows:

```
DLHCSYSE.DMS
```

```
DLHCSYSI.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	Oracle	All

## Task 4-14-12: Exporting Application System Data

This script exports system data from various application tables from the Demo database into a Data Mover \*.DAT file. In a later step, this data will be loaded into the Copy of Production database. The script name for your upgrade path is:

```
DLHCSYSE.DMS
```

---

**Note.** During Move to Production passes, you can reuse the data files that are created by this export script. To do this, change the Type of Upgrade from Both to Initial Upgrade in the step properties and save the job.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 4-14-13: Importing Application System Data

This script imports the application system data, exported in the previous step, into the Copy of Production database. The script name for your upgrade path is:

```
DLHCSYSI.DMS
```

---

**Note.** Some of the data will be imported using the *ignore dups* option. These data loads will give the message “Error: duplicate SQL rows” and then give a “Successful completion” message. These error messages can be ignored because duplicate data is expected.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-14-14: Exporting Data for Your Upgrade Path

This script exports system data from application tables from the Demo database into a Data Mover \*.DAT file that is needed for upgrading from your current release. In a later step, this data will be loaded into the Copy of Production database. The script name for your upgrade path is:

DLHCUPG05E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

### Task 4-14-15: Importing Data for Your Upgrade Path

This script imports the application system data that is needed for upgrading from your current release, which was exported in the above step, into the Copy of Production database. The script name for your upgrade path is:

DLHCUPG05I.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-14-16: Exporting Common Portal System Options

This script exports the contents of the Common Portal System Options table from the Demo database. The script name for your upgrade path is:

DLEOX01E.DMS

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All



## Task 4-14-17: Importing Common Portal System Options

This script imports the Common Portal System Options data into your Copy of Production database. The script name for your upgrade path is:

`DLEOX01I.DMS`

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-14-18: Running the DLHCCOG03E Script

This script exports additional system data from various application tables from the Demo database into a Data Mover \*.DAT file. This data will be loaded into the Copy of Production database in a later step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 4-14-19: Running the DLHCCOG03I Script

This script imports the additional application system data, exported in the above, into the Copy of Production database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-14-20: Exporting Setup Data

This script exports setup data from the Demo database. The script name for your upgrade path is:

`DLUPX16E.DMS`

This data will be exported during Move to Production by the script MVAPPEXP.DMS.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Initial	All	All	All

## Task 4-14-21: Importing Setup Data

This script imports setup data into the Copy of Production database. The script name for your upgrade path is:

```
DLUPX16I.DMS
```

This data will be imported during Move to Production by the script MVAPPIMP.DMS.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 4-14-22: Exporting Approval Framework Definitions

This script exports Approval Framework transaction definitions from the Demo database. The script name for your upgrade path is:

```
DLHCUPW10E.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

## Task 4-14-23: Importing Approval Framework Definitions

This script imports Approval Framework transaction definitions from the Demo database. The script name for your upgrade path is:

```
DLHCUPW10I.DMS
```

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-14-24: Exporting Generic Notifications

This step exports Generic Notification Templates data from the Demo database. The script name for your upgrade path is:

```
DLHCUPT04E.DMS
```

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	Both	All	All	All

**Task 4-14-25: Importing Generic Notifications**

This step imports Generic Notification Templates data into your Copy of Production database.

The script name for your upgrade path is:

DLHCUPT04I.DMS

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

**Task 4-15: Loading Stored Statements**

This section discusses:

- Understanding Loading Stored Statements
- Setting PSOPTIONS for COBOL
- Running the STOREFA Script
- Running the STORECC Script
- Running the STOREAD Script
- Running the STOREAV Script
- Running the STORESF Script
- Running the STORESR Script
- Running the STOREBAS Script
- Running the STOREGP Script
- Running the STOREHRM Script
- Running the STOREPAY Script
- Running the STOREPEN Script
- Running the STOREPYI Script

## Understanding Loading Stored Statements

This task loads all the new COBOL stored statements in your database. The STORE\*.DMS scripts are located in the PS\_HOME\SRC\CBL\BASE directory. These scripts are only delivered if you have licensed the product and should only appear in the Change Assistant job based on the product configuration in the environment definition.

The stored statements scripts are executed only in the initial pass. In the move to production passes, a different script will export the stored statement table, PS\_SQLSTMT\_TBL, from the first copy of production and import it into the second copy of production.

If you use DB2 z/OS you also update the COBOL settings in the PSOPTIONS in this task.

### Task 4-15-1: Setting PSOPTIONS for COBOL

This step runs DB2ALLCCSIDUPD.SQL and updates PSOPTIONS.DB2ALLCCSIDOK. The purpose of the field PSOPTIONS.DB2ALLCCSIDOK is to control whether COBOL processing should be allowed for COBOL processes running under Windows NT/2000 or UNIX against a DB2 z/OS database where the DB2 subsystem zparm for SCCSID is not 37 (US English EBCDIC). A value of 'N' in this field will result in the COBOL process being terminated and value of 'Y' will allow the process to continue.

---

**Note.** Move to Production: PSOPTIONS is copied in the Move to Production pass with the MVPRDEXP/IMP scripts.

---

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	DB2 z/OS	All

### Task 4-15-2: Running the STOREFA Script

STOREFA.DMS loads stored statements for COBOL programs owned by the Financial Aid Product.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

### Task 4-15-3: Running the STORECC Script

STORECC.DMS loads stored statements for COBOL programs owned by the Campus Community product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

**Task 4-15-4: Running the STOREAD Script**

STOREAD.DMS loads stored statements for COBOL Programs owned by the Recruiting and Admissions product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

**Task 4-15-5: Running the STOREAV Script**

STOREAV.DMS loads stored statements for COBOL programs owned by the Contributor Relations product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration Contributor Relations	All	All

**Task 4-15-6: Running the STORESF Script**

STORESF.DMS loads stored statements for COBOL programs owned by the Student Financials product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

**Task 4-15-7: Running the STORESR Script**

STORESR.DMS loads stored statements for COBOL programs owned by the Student Records and Academic Advisement products.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

**Task 4-15-8: Running the STOREBAS Script**

In this step, you run STOREBAS.DMS. This script loads stored statements for COBOL programs owned by the Benefits Administration product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Benefits Administration	All	All

**Task 4-15-9: Running the STOREGP Script**

In this step, you run STOREGP.DMS. This script loads stored statements for COBOL programs owned by the Global Payroll product.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Global Payroll Core	All	All

**Task 4-15-10: Running the STOREHRM Script**

In this step you run STOREHRM.DMS. This script loads stored statements for COBOL programs.

**Properties**

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

**Task 4-15-11: Running the STOREPAY Script**

In this step, you run STOREPAY.DMS. This script loads stored statements for COBOL programs owned by the Payroll for North America product.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

## Task 4-15-12: Running the STOREPEN Script

In this step, you run STOREPEN.DMS. This script loads stored statements for COBOL programs owned by the Pension Administration product.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Pension Administration	All	All

## Task 4-15-13: Running the STOREPYI Script

In this step, you run STOREPYI.DMS. This script loads stored statements for COBOL programs owned by the Payroll Interface product.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll Interface	All	All

---

## Task 4-16: Running Final Update Statistics

This section discusses:

- Generating Final RUNSTATS for DB2 UNIX
- Running Final Statistics for DB2 UNIX
- Running Final Statistics for DB2 zOS
- Running Final Statistics for Informix
- Running Final Statistics for Oracle

### Task 4-16-1: Generating Final RUNSTATS for DB2 UNIX

This step executes the RUNSTATS.SQR that creates the RUNSTATS.SQL to update statistics on DB2 UNIX/NT.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX/NT	All

### Task 4-16-2: Running Final Statistics for DB2 UNIX

Earlier in the upgrade process you updated your statistics. Now that you have converted all of your data and modified all indexes, update your statistics again to improve performance of your post upgrade processes and testing. Run the RUNSTATS.SQL script created in the previous step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 UNIX	All

### Task 4-16-3: Running Final Statistics for DB2 zOS

Earlier in the upgrade process you updated your statistics. Now that you have converted all of your data and modified all indexes, update your statistics again to improve performance of your post upgrade processes and testing. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS	All

### Task 4-16-4: Running Final Statistics for Informix

Earlier in the upgrade process you updated your statistics. Now that you have converted all of your data and modified all indexes, update your statistics again to improve performance of your post upgrade processes and testing. This step runs UPDATESTATS to update the statistics on your database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Informix	All



## Task 4-16-5: Running Final Statistics for Oracle

Earlier in the upgrade process you upgraded your statistics. Now that you have converted all of your data and modified all indexes, update your statistics again to improve performance of your post upgrade processes. Contact your database administrator to have the statistics updated on your database before proceeding with your upgrade and testing.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All

---

## Task 4-17: Completing Application Processes

This section discusses:

- Refreshing Personal Data Table
- Running Data Conversion for Group 20
- Exporting Setup Data for Rules
- Importing Setup Data for Rules
- Exporting Payroll Interface Tables
- Importing Payroll Interface Tables
- Exporting GL Interface Setup Tables
- Importing GL Interface Setup Tables
- Exporting US Custom Garnishment Rules
- Importing US Custom Garnishment Rules

### Task 4-17-1: Refreshing Personal Data Table

In this step, you refresh the Personal Data snapshot table used for reporting. Run the Application Engine program PERS\_REFRESH to refresh the table.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-17-2: Running Data Conversion for Group 20

This step runs the UPG\_DATACONV Application Engine program for Group 20. If you want to see additional documentation for Group 20, then run the UDATACNV report.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-17-3: Exporting Setup Data for Rules

Run this step only during Move to Production passes. It exports data from tables that were updated for Template Built and Custom Rules setup during the initial pass.

Run the following script:

```
MVTLW01E.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	Time and Labor	All	All

## Task 4-17-4: Importing Setup Data for Rules

Run this step only during Move to Production passes. It will import data from tables that were updated for Template Built and Custom Rules setup during the initial pass.

Run the following script:

```
MVTLW01I.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	Time and Labor	All	All

## Task 4-17-5: Exporting Payroll Interface Tables

This step is only run during the Move to Production passes. It exports all Payroll Interface definitions including data from tables that were updated in the chapter “Complete Database Changes,” Updating Payroll Interface Definitions task during the initial pass.

The script name for your path is:

```
MVHCPIT01E.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	Payroll Interface	All	All

## Task 4-17-6: Importing Payroll Interface Tables

This step is only run during the Move to Production passes. It imports all Payroll Interface definitions including data from tables that were updated in the chapter “Complete Database Changes,” Updating Payroll Interface Definitions task during the initial pass.

The script name for your path is:

```
MVHCPIT01I.DMS
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	Payroll Interface	All	All

## Task 4-17-7: Exporting GL Interface Setup Tables

In this step, you export General Ledger Interface setup tables. This step should only be run if you use non-commitment accounting. If you use commitment accounting, mark this step as complete and continue your upgrade, since you could lose data that was entered between upgrade passes if you run this step.

To set this step to run:

1. In your job in Change Assistant, highlight this step and select Edit, Step Properties.
2. In the Step Properties dialog box, change the Type from *Manual* to *DataMoverUser*.
3. Click OK.

This step is only run during the Move to Production passes. If you are using non-commitment accounting in Payroll for North America, you will set up data in tables used for General Ledger Interface during the initial pass. This script will export the data from these tables.

The script name for your path is:

```
MVHCPYW01E.DMS
```

See Appendix: “Understanding Dynamic ChartFields Changes.”

See “Complete Database Changes,” Setting Up General Ledger Interface.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	Payroll for North America	All	All

## Task 4-17-8: Importing GL Interface Setup Tables

In this step, you import General Ledger Interface setup tables. This step should only be run if you use non-commitment accounting. If you use commitment accounting, mark this step as complete and continue your upgrade, since you could lose data that was entered between upgrade passes if you run this step.

To set this step to run:

1. In your job in Change Assistant, highlight this step and select Edit, Step Properties.
2. In the Step Properties dialog box, change the Type from *Manual* to *DataMoverBootstrap*.
3. Click OK.

This step is only run during the Move to Production passes. If you are using non-commitment accounting in Payroll for North America you will setup the data in tables used for General Ledger Interface during the initial pass. This script will import the data from these tables.

The script name for your path is:

```
MVHCPYW01I.DMS
```

See Appendix: “Understanding Dynamic ChartFields Changes.”

See “Complete Database Changes,” Setting Up General Ledger Interface.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	Payroll for North America	All	All

## Task 4-17-9: Exporting US Custom Garnishment Rules

This step is only run during the Move to Production passes. If you have created any U.S. garnishment rules, this script will export those rules from garnishment rules tables.

---

**Note.** If you have not created any U.S. garnishment rules, you do not have to run this task.

---

The script name for your path is:

```
MVHCPYT05E.DMS
```

See Appendix: “Understanding Garnishments Changes.”

See “Complete Database Changes,” Setting Up US Custom Garnishment Rules.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Source	MTP	Payroll for North America	All	All

## Task 4-17-10: Importing US Custom Garnishment Rules

This step is only run during the Move to Production passes. If you have created any U.S. garnishment rules and run the export script, this script will import those rules into garnishment rules tables.

---

**Note.** If you have not created any U.S. garnishment rules, you do not have to run this task.

---

The script name for your path is:

`MVHCPYT05I.DMS`

See Appendix: “Understanding Garnishments Changes.”

See “Complete Database Changes,” Setting Up US Custom Garnishment Rules.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	Payroll for North America	All	All

## Task 4-18: Updating Language Data

This section discusses:

- Understanding Updating Language Data
- Run the TSRECPOP Script

## Understanding Updating Language Data

In this task, you run scripts to modify data in PeopleTools related language tables.

---

**Note.** For DB2 z/OS customers, PeopleSoft recommends that you run RUNSTATS against the system catalog tables at this time.

---

## Task 4-18-1: Run the TSRECPOP Script

In this step, the TSRECPOP script initializes and modifies the data in PeopleTools related language architecture tables.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 4-19: Updating Object Version Numbers

In this task, you run the VERSION Application Engine program. This ensures that all of your version numbers are correct and, if not, resets them to 1.

---

**Note.** Do not update statistics after you complete this task.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 4-20: Running the Final Audit Reports

This section discusses:

- Run the Final DDDAUDIT Report
- Run the Final SYSAUDIT Report
- Create the FNLALTAUD Project
- Run the Final Alter Audit
- Review the Final Audits
- Run a Final SETINDEX Report
- Run a Final SETTABLE Report

### Task 4-20-1: Run the Final DDDAUDIT Report

DDDAUDIT is an SQR that compares your production SQL data tables with the PeopleTools record definitions to uncover inconsistencies. You can expect some errors from this report. You will review the output from the report in another step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-20-2: Run the Final SYSAUDIT Report

SYSAUDIT is an SQR that identifies *orphaned* PeopleSoft objects. For example, SYSAUDIT will identify a module of PeopleCode that exists but does not relate to any other objects in the system. SYSAUDIT also identifies other inconsistencies within your database.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-20-3: Create the FNLALTAUD Project

In this step, you create the FNLALTAUD project and use it to run your final Alter Audit. Creating this new project now ensures that all the records in your system are audited, including SQL tables. This project also includes any custom records that you have created in your system.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 4-20-4: Run the Final Alter Audit

Run the PeopleTools alter record process on all tables in your system to check whether the PeopleTools definitions are synchronized with the underlying SQL data tables in your database. PeopleSoft calls this process an Alter Audit. Alter Audit compares the data structures of your database tables with the PeopleTools definitions to uncover inconsistencies. Alter Audit then creates an SQL script with the DDL changes needed to synchronize your database with the PeopleTools definitions.

The Alter Audit script is built using the FNLALTAUD project created in the previous step.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-20-5: Review the Final Audits

The Alter Audit process creates SQL scripts that correct any discrepancies between your PeopleTools record definitions and the database system catalog table definitions. Review the Alter Audit output and correct any discrepancies noted by running the generated scripts with your platform-specific SQL tool. The script names are:

```
FNLALTAUD_ALTTBL.SQL
FNLALTAUD_CRTIDX.SQL
```

---

**Note.** The Alter Audit process also creates the script FNLALTAUD\_CRTTRG.SQL, which re-creates all database triggers. You do not need to run this script, since all database triggers were created in a previous task.

---

See Finalizing the Database Structure.

---

**Note.** For Informix sites, if your database has Application Functions, you use SQL to drop and re-create these functions and their associated indexes, even though the underlying tables and indexes have not changed.

---



---

**Note.** For Microsoft SQL Server and DB2 UNIX platforms, if your database has tables containing the MSSCONCATCOL or DBXCONCATCOL column, you will see SQL alter the tables and re-create their associated indexes, even though the underlying tables and indexes may not have changed.

---

Review the output from the SYSAUDIT and DDDAUDIT reports and correct any discrepancies.

Your DDDAUDIT listing shows some expected discrepancies. Tables and views deleted from the Application Designer are not automatically deleted from the system tables. PeopleSoft takes this precaution in case you have customized information that you want to preserve. Therefore, the report lists any tables and views that the new release does not have. Review these tables to verify that you do not wish to preserve any custom data, and then drop the tables and views.

Your SYSAUDIT report may have some errors due to references to obsolete PeopleSoft-owned objects. For instance, if PeopleSoft deletes a Permission List, and you have a Role that still refers to that Permission, then it will appear on the SYSAUDIT report.

See the Enterprise PeopleTools PeopleBook: Data Management for your new release.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 4-20-6: Run a Final SETINDEX Report

The SETINDEX SQR updates index overrides stored in the PSIDXDDLPRM table. The SQR updates the values stored in the PARMVALUE field with current values found in the system catalog. Running SETINDEX cleans up fragmentation issues that may have occurred during data conversion.



## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	DB2 z/OS Oracle	All

## Task 4-20-7: Run a Final SETTABLE Report

The SETTABLE SQR updates table overrides stored in the PSRECDDLPARM table. The SQR updates the values stored in the PARMVALUE field with the current values found in the system catalog. Running SETTABLE will clean up fragmentation issues that may have occurred during data conversion.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	Oracle	All



## CHAPTER 5

# Complete Database Changes

This chapter discusses:

- Understanding Database Changes
- Configuring the Upgrade Environment
- Reapplying Customizations
- Setting Up Security
- Updating Department Security
- Backing Up Before Manual Changes
- Converting Retirees with Pay
- Verifying Equation Conversion
- Running the ECPrep Process
- Running the Person Organization Audits
- Setting Up Pension
- Reviewing the Benefits Limit Table
- Setting Up Stock Administration
- Validating Alternative Overtime Upgrade
- Updating Payroll Interface Definitions
- Upgrading Rules
- Deleting Resume Files
- Setting Up General Ledger Interface
- Validating EE Garn Payee Data
- Setting Up US Custom Garnishment Rules
- Validating EE Garn Rule Data
- Updating the Employee ID Length
- Setting Up Student Group Security
- Reviewing PeopleTools Functionality
- Preparing the Content Provider Registry
- Updating the Portal Options Data
- Stamping the Database

- Reviewing Change Control
- Backing Up Before Testing
- Testing Your Copy of Production

---

## Understanding Database Changes

Many changes were made in the previous chapters of this documentation. In this chapter, you complete these changes so that you can begin testing your Copy of Production. By testing your Copy of Production, you ensure that you can still operate day-to-day processes on your new PeopleSoft release.

---

### Task 5-1: Configuring the Upgrade Environment

This section discusses:

- Configure Application Server
- Configure Portal

#### Task 5-1-1: Configure Application Server

Running Portal requires a fully functional application server domain. In this step, you configure your application server.

---

**Note.** If you configured your application server earlier in the upgrade, you can skip this step.

---

Tips for configuring and starting the application server:

- Make sure that the Application Server domain you configure points to the Target database for this pass of the upgrade.
- Set a different JSL port for each database instance.

See the Enterprise PeopleTools installation guide for your database platform.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

#### Task 5-1-2: Configure Portal

PeopleSoft applications are accessed through the Portal. You need to grant users access to complete the upgrade process. You must install and configure the Portal to complete the upgrade.

---

**Note.** If you configured your Portal earlier in the upgrade, you can skip this step.

---

You also must define a password on the Node Definitions page for Single Signon to work properly. If you do not define a password, the signon page appears when trying to access a report directly, instead of the report itself. To avoid this issue, follow the procedure below to assign a password.

To assign a password:

1. Select PeopleTools, Integration Broker, Integration Setup, Nodes.
2. Click Search.
3. Select the database's default local node.  
The default local node shows a *Y* in the Default Local Node column.
4. On the Node Definitions page, select *Password* in the Authentication Option field.
5. Enter a password in the Password field.
6. Enter the password again in the Confirm Password field.
7. Enter the default user in the Default User ID field.
8. Save the node definition.
9. Reboot the application server and web server.

See the Enterprise PeopleTools installation guide for your database platform.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-2: Reapplying Customizations

This section discusses:

- Understanding the Reapplication
- Performing Customized Object Adjustment
- Registering Portal Navigation Objects

## Understanding the Reapplication

In this task, you work with your customized objects to ensure that they are properly integrated into your upgraded database.

When reapplying customizations to batch processes, please review the appendix “Reviewing Batch Program Changes” for information about changes in batch processes.

## Task 5-2-1: Performing Customized Object Adjustment

When you reviewed your upgrade compare reports, you decided whether to take the Source or Target version of the objects. If you took the Oracle-delivered version of an object instead of your own customized version, you may need to customize the new objects to get the blend of new standard features and your custom features. In complex cases, this may take several iterations. You need to make manual adjustments to the objects to apply these customizations.

Once you reapply all of your customizations, you should run the DDDAUDIT and SYSAUDIT reports to make sure that you did not introduce any problems into your system.

Reapply any Mass Change or EDI customizations.

See “Prepare Your Database,” Identifying Customizations.

Be aware that you must not overwrite Oracle-loaded data. The customizations, extracted during an earlier step, must be manually applied now.

In another step, you applied the Oracle-delivered record group assignments.

See “Apply Application Changes,” Loading Data for Data Conversion, Importing Record Groups.

If you maintain any custom record group assignments, reapply them to your Copy of Production database now.

During Move to Production passes, you will not need to reapply these customizations. The changes you make now will be copied to any subsequent Copy of Production database using Data Mover scripts.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 5-2-2: Registering Portal Navigation Objects

You must register your customized objects, such as menus and components, in order to access them in Portal. You can use the Registration Wizard or the Menu Import process to grant access to the appropriate components. Make sure that you register your components for all of your portals (for example, Customer, Supplier, Employee, and so forth). Also, make sure that you select the node name that matches the database. Do not use the Local node.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Application Designer for your new release, “Using the Registration Wizard.”

See the Enterprise PeopleTools PeopleBook: Internet Technology for your new release, “Administering Portals.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 5-3: Setting Up Security

This section discusses:

- Understanding Security
- Set Up Security
- Synchronize CREF Permissions
- Grant Access to Personalize Homepage

### Understanding Security

In this task you perform steps to set up security, grant access to the user ID, set up permissions lists, and grant access to navigation and homepages.

### Task 5-3-1: Set Up Security

This section discusses:

- Understanding Security Setup

#### Understanding Security Setup

Select the PeopleTools, Security folder now to add the new PeopleTools and application menus, delete old menus, and set up appropriate operator security for your system.

Many menu additions and deletions have occurred. Examine the menu compare report and the PeopleSoft Demo database for details of the required security changes, then decide which of your roles and permission lists should have access to each of the new menus.

Many tasks in this chapter instruct you to select a specific menu within the new PeopleSoft release. To perform these tasks, set up appropriate security for each of the menus referenced in each of the tasks.

At this time, you need to review the two security views that are used for Global Security and Row Level Security. The following naming conventions have been adopted in the new PeopleSoft HRMS release:

List Type	Format
Data Permission	HCDPXXXXXX (Where XXXXXX can be any specification you choose.)
Component Permission Lists	HCCPXXXXXX (Where XX is the product code and YYYY is any specification you choose.)
Primary Permission Lists	HCPPXXXXXX (where XXXXXX is any specification you choose.)

If you decide to change your security to use this naming convention, you do not need to take any further action. However, if you do not intend to use the naming convention given above, then you need to modify the WHERE clause in the following two views to reference the naming convention you have decided to follow.

Security View	Function
OPRDEFN_SCRTY3	Data Permissions (Row Level Security)
OPRDEFN_SCRTY4	Primary Permissions (Global Security)

See the PeopleSoft Enterprise Portal Solutions PeopleBook: Enterprise Portal Application Technology for your new release, information on Oracle-delivered security.

**Note.** Move to Production: If you changed the user profiles in your production system after you froze your PeopleTools, you must manually apply the changes to your Copy of Production database before the end of the final Move to Production.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 5-3-2: Synchronize CREF Permissions

This section discusses:

- Understanding Content Reference Permissions
- Running the Portal Security Synchronization Process

### Understanding Content Reference Permissions

As part of the PeopleTools 8.4x Portal architecture, Portal Registry Structures reference permission lists. At this point, however, the Portal Registry Structures copied from the Demo database do not reference any permission lists on the Copy of Production database. This synchronization program will match the existing permission lists to the appropriate Registry Structures and update it.

**Note.** The user ID that invokes this process must have the security role Portal Administrator. Otherwise, the process may terminate abnormally.

**Note.** Your Process Scheduler must be running in order to perform this task.

### Running the Portal Security Synchronization Process

Follow the steps below to run the Portal security synchronization process.

To run the security synchronization process:

1. From your browser, sign in to your Target database.
2. Select PeopleTools, Portal, Portal Security Sync.
3. Click Add a New Value.
4. Enter the run control ID *UPG\_PORTAL\_SYNC\_BOTH*.
5. Click Add.



6. Keep the default value for the default portal registry name in the Portal Name field (for example: *EMPLOYEE*, *CUSTOMER*, or *SUPPLIER*).
7. Click Save.
8. Click Run.
9. In the Process Scheduler page, check that you set your parameters correctly.
10. Click OK.
11. Click the Process Monitor link to monitor the program's process.
12. Repeat steps 6 through 11 for each Portal name used in the database for your specific applications.

With each repetition, in step 6 change the Portal Name field to one of the following: *EMPLOYEE*, *CUSTOMER*, *SUPPLIER*, *MOBILE*, and so on.

13. Review any messages received during the running of this process with your Portal Administrator.

See the Enterprise PeopleTools PeopleBook: Internet Technology for your new release.

---

**Note.** If the permission lists for your upgrade user do not allow you access to a component, you will encounter this error when running the security synchronization process for that page: `Security synchronization failed for Portal Object`. This error may indicate other problems with the component or folder, but you should check your security first.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 5-3-3: Grant Access to Personalize Homepage

This section discusses:

- Understanding Access to Portal Homepage
- Updating Homepage Personalization Permission List
- Adding the Portal User Role

### Understanding Access to Portal Homepage

You must complete this step if you use any of the Portal Pack products or pagelets. In order to add, remove, or change the layout of the homepage, you must grant homepage personalization security access to all non-guest users.

### Updating Homepage Personalization Permission List

To update the homepage personalization permission list:

1. Using PeopleSoft Data Mover, sign in to the Target database.
2. Open the Data Mover script *PS\_HOME\SCRIPTS\PORTAL\_HP\_PERS.DMS*.
3. Run this script against the Target database.
4. Close Data Mover.

## Adding the Portal User Role

To add the Portal User Role to the user IDs:

1. Using PeopleSoft Data Mover, sign in to the Target database.
2. Open the Data Mover script *PS\_HOME\SCRIPTS\PORTAL\_ADD\_ROLE.DMS*.
3. Run this script against the Target database.
4. Close Data Mover.

---

**Note.** You should grant the PAPP\_USER role to all new user IDs for access to the homepage personalization. After running this script, manually remove the role PAPP\_USER from any GUEST user ID, because a GUEST user should not be personalizing the common homepage.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 5-4: Updating Department Security

This section discusses:

- Understanding Department Security
- Refreshing Operator Security
- Refreshing Transaction Records

## Understanding Department Security

In this task you update the Operator Security Join table and the transaction side Security Join Tables so that they are based on the most current information. You need to run this step whether you use department level security or not.

### Task 5-4-1: Refreshing Operator Security

This process refreshes the Operator Security Join Table. You must run this process whenever a Security Type is modified, when a Security Tree is modified or added, or when a ROWSECCLASS is modified or added.

To refresh the Operator Security Join Table:

1. From your browser, sign on to your Copy of Production database, keeping the default browser address.
2. Select Set Up HRMS, Security, Core Row Level Security, Refresh SJT\_CLASS\_ALL.
3. Click Add a New Value.
4. Enter *UPGR\_OPRSECURITY* on the Run Control selection panel and click Add.
5. On the Run Control page, keep all preset defaults for upgrading and click Run.

6. On the Process Scheduler Request page, click OK.
7. Monitor the process from the Process Monitor.
8. Select Set Up HRMS, Security, Core Row Level Security, Refresh SJT\_OPR\_CLS.
9. Enter *UPGR\_OPRSECURITY* on the Run Control selection panel and click Search.
10. On the Run Control page, keep all preset defaults for upgrading and click Run.
11. On the Process Scheduler Request page, click OK.
12. Monitor the process from the Process Monitor

See *PeopleSoft Enterprise HRMS 9.0 Application Fundamentals*, “Setting Up and Administering HRMS Security.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-4-2: Refreshing Transaction Records

This process updates the transaction side Security Join Tables.

To refresh the transaction side Security Join Tables:

1. From your browser, sign on to your Copy of Production database, keeping the default browser address.
2. Select Set Up HRMS, Security, Core Row Level Security, Refresh Trans SJT Tables.
3. Click Add a New Value.
4. Enter *UPGR\_OPRSECURITY* on the Run Control selection panel and click Add.
5. On the Run Control page, keep all preset defaults for upgrading and click Run.
6. On the Process Scheduler Request page, click OK.
7. Monitor the process from the Process Monitor.

See *PeopleSoft Enterprise HRMS 9.0 Application Fundamentals*, “Setting Up and Administering HRMS Security.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-5: Backing Up Before Manual Changes

Back up your Copy of Production database now. This enables you to restart your upgrade from this point should you experience any database integrity problems during the remaining tasks in the upgrade process.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-6: Converting Retirees with Pay

This section discusses:

- Understanding the Conversion for Retirees with Pay
- Running Retiree Additional Pay Report UVPYT05
- Running Retiree Additional Pay Report UVPYT06
- Running Retiree Additional Pay Report UVPYT07
- Running Upgrade Conversion for Retirees with Pay
- Rebuilding Job Flags

### Understanding the Conversion for Retirees with Pay

In this task, you migrate retirees receiving a pension outside of Pension Administration from employees to persons of interest, in order to follow the new Person Model design.

In releases prior to PeopleSoft 8.9, retirees that were paid through the system but outside of Pension Administration appeared as employees. With the Person Model changes delivered in PeopleSoft 8.9 and beyond, these retirees become Persons of Interest (POI). Because there is no way to predict every scenario used to process the pension for the retiree, we must deliver this conversion outside of the standard upgrade data conversion. If your data fits the scenario outlined in the list below, then you should complete this step.

---

**Note.** If your data does not fit the scenario outlined below, please consult the Global Support Center (GSC) to obtain a solution specific to your business requirements.

---

- This conversion assumes that these retirees belong to a company with a Tax Report Type of *R* for 1099R.
- This conversion assumes that your Payroll System is Payroll for North America
- This conversion assumes that you are *not* using Benefits Billing.
- This conversion assumes that you are *not* using Benefits Administration Credits.
- If the EMPL\_RCD value for the employment instance is the same value for the retirement instance, the conversion sets the EMPL\_RCD value to the next value available for the person on the POI instance.

This includes changing the value on the payroll tables.

- If the retiree transfers to the 1099R company, the conversion changes the transfer (XFR) ACTION to Retired with Pay (RWP).

The conversion then copies the data from the row to insert the POI instance.

- If the retiree terminates from the actual company and hires into the 1099R company, the conversion changes the termination (TER) ACTION to Retired with Pay (RWP).

The conversion also changes the hire (HIR) ACTION into the 1099R company to Add Person of Interest (POI).

- This conversion sets the Benefit System for the retiree to the value specified on the Upgrade Retiree Run Control page.
- This conversion will insert or update the following tables:

- Human Resources

PER\_ORG\_ASGN - Insert for Retirees, Update for Survivor Beneficiaries & Joint⇒  
Annuitants.

PER\_ORG\_INST -Insert

JOB - Update and Insert if retiree transfers to 1099R Company

JOB\_JR - Update and Insert if retiree transfers to 1099R Company

JOB\_EARNS\_DIST - Update

COMPENSATION - Update

- Federal

GVT\_JOB - Update & Insert if Retiree transfers to 1099R Company

GVT\_EE\_KEY - Insert

GVT\_PERS\_DATA - Update & Insert if Retiree transfers to 1099R Company

GVT\_PERS\_PHONE - Update

GVT\_PERS\_NID - Update

GVT\_EMPLOYMENT - Update

GVT\_CITIZENSHIP - Update

GVT\_DISABILITY - Update

GVT\_JOBDIST - Update

GVT\_AWD\_DATA - Update

GVT\_EE\_DATA\_TRK - Update

JOB\_USF - Update

GVT\_PAYROLL\_DATA - Update

GVT\_PAY\_LIMITS - Update

### - Payroll

TAX\_DIST\_EFFDT - if row exists Update else Insert.  
 TAX\_DISTRIB - if row exists Update else Insert  
 ADDL\_PAY\_DATA - See Below  
 ADDL\_PAY\_EFFDT - See Below  
 ADDL\_PAY\_ERNCD - See Below  
 BAL\_ADJ\_CN\_ERN - Update  
 BAL\_ADJ\_ERN - Update  
 CAN\_ERN\_BALANCE - Update  
 DED\_LINE - Update  
 EARNINGS\_BAL - Update  
 EMPL\_WAGELS\_CAN - Update  
 PAY\_CHECK - Update  
 PAY\_EARNINGS - Update  
 PAY\_LINE - Update  
 PSHUP\_TXN - Update  
 PAY\_SPCL\_EARNS - Update.

- For the scenarios outlined below, the conversion will convert the ADDL\_PAY\_ERNCD, ADDL\_PAY\_EFFDT, and ADDL\_PAY\_DATA tables in the manner described.
  - The maximum EFFDT with End Date is before the retirement date.  
 All rows are applicable to the employee's history. No action is taken
  - The maximum EFFDT with End Date is after the retirement date.  
 Some rows are applicable to the employee's history, while some are applicable to the retirement history. This data is too cumbersome to convert, therefore we recommend that you convert this data manually. We provide an SQR Report, UVPYT05.SQR, to identify these rows.
  - The maximum EFFDT is before the retirement date.  
 This history is applicable to both employment and retirement periods. Therefore, the conversion process will insert a row for the retirement data. We also provide an SQR Report, UVPYT06.SQR to identify these rows.
  - The maximum EFFDT is after the retirement date.  
 This history is applicable to both employment and retirement periods. Therefore, the conversion process will insert a row for the retirement data. We also provide an SQR Report, UVPYT07.SQR to identify these rows.
  - The minimum EFFDT is after the retirement date.  
 This data is only applicable to the retiree's history. Therefore, the conversion process will update these rows, setting the EMPL\_RCD value to the value of the POI instance.

- This conversion does *not* modify the following tables:

#### Payroll

AP\_EXTRACT\_LINE - if you use AP to create vouchers the data here will not be⇒  
 consistent since we cannot update the data. However, you will be able to view⇒  
 the information by navigating to: Payroll for North America > Pay Distribution⇒  
 > Accounts Payable Information > Review AP Extract Lines

WRK\_CHECK - you must complete Payroll processing before upgrading.  
 WRK\_EARNINGS - <same as above>  
 WRK\_LINE - <same as above>  
 WEK\_PSHUP\_TXN - <same as above>  
 WRK\_SPCL\_EARNS - <same as above>  
 PYRE\_DETAIL - <same as above>  
 PYRE\_INVALID <same as above>  
 TL\_EMPL\_DATA - not expecting Retirees to use Time & Labor  
 PY\_LDTLEMP\_TBL - <same as above>  
 CSB\_EMPL\_SERIES - obsolete Canadian Savings Bond functionality  
 CSB\_REGISTRANT - <same as above>  
 CSB\_REG\_DEMON - <same as above>

## Task 5-6-1: Running Retiree Additional Pay Report UVPYT05

In this step, you run the Retiree Additional Pay SQR Report, UVPYT05.SQR. This report lists retirees with pay that have additional pay with an end date where the maximum EFFDT is after the retirement date.

To run the report:

1. From your web browser, sign in to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, 1099R AddlPay after Ret w/ End.
3. Select Add a New Value
4. Enter *UPG\_RET* for the Run Control ID.
5. Click Add.
6. On the Add'l Pay after Ret w/ End page, click Run.
7. From the Process Scheduler Request page, click OK.
8. Analyze the report to decide what rows belong to the retirement history for the each person.

Apply manual modifications after running the conversion by setting the retirement rows to the EMPL\_RCD value assigned by the conversion for the POI instance.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-6-2: Running Retiree Additional Pay Report UVPYT06

In this step, you run the Retiree Additional Pay SQR Report, UVPYT06.SQR. This report lists retirees with pay that have additional pay *without* an end date where the maximum EFFDT is before the retirement date.

To run the report:

1. From your web browser, sign in to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, 1099R AddlPay b4 Ret w/o End.

3. Enter *UPG\_RET* for the Run Control ID.
4. Click Search.
5. On the Add'l Pay b4 Ret w/o End page, click Run
6. From the Process Scheduler Request page, click OK.
7. Keep this report as a reference to verify that the conversion converted the data properly.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-6-3: Running Retiree Additional Pay Report UVPYT07

In this step, you run the Retiree Additional Pay SQR Report, UVPYT07.SQR. This report lists retirees with pay that have additional pay *without* an end date where the maximum EFFDT is after the retirement date

To run the report:

1. From your web browser, sign in to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, 1099R AddlPay after Ret w/o End.
3. Enter *UPG\_RET* for the Run Control ID.
4. Click Search.
5. On the Add'l Pay after Ret w/o End page, click Run.
6. From the Process Scheduler Request page, click OK.
7. Keep this report as a reference to verify that the conversion converted the data properly.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-6-4: Running Upgrade Conversion for Retirees with Pay

In this step, you run the Application Engine Program to upgrade retirees with pay from employees to persons of interest.

To run the program:

1. From your web browser, sign in to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, 1099R Conversion.
3. Select Add a New Value.
4. Enter *UPG* for the Run Control ID.
5. Click Add.



- In the Benefits System field, select a value for retirees with pay.  
The JOB table will be populated with this value.

Upg Ret Runcntl page

- Click the Run button.
- From the Process Scheduler Request page, click OK.
- Click the Process Monitor link to monitor the program's process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-6-5: Rebuilding Job Flags

After successfully completing the conversion in the previous step, you must rebuild PRIMARY\_JOBS by running the SQR utility BEN500.SQR.

To run the SQR utility:

- From your web browser, sign in to the Copy of Production database.
- Select Benefits, Maintain Primary Jobs, Rebuild Primary Job Flags.
- Select Add a New Value.
- Enter the run control *UPG\_PRIMARY\_JOBS* and click Run.
- Click the Process Monitor link to monitor the program's process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-7: Verifying Equation Conversion

This section discusses:

- Understanding the Equation Conversion
- Running the Equation Conversion Reports
- Reviewing Manual Conversion Candidates
- Reviewing Equations
- Reviewing Equation Security

### Understanding the Equation Conversion

During data conversion, equations from PKG\_EQTN\_TBL and SF\_PKG\_EQTN\_TBL were converted. If the equations did not follow expected patterns, assumptions were made in order to complete the conversion. In this task, you will run a report that lists all of these assumptions and review the converted equations to make sure they work as expected.

---

**Note.** Your equations will only be converted during the initial pass. Any changes you make in this task will be brought forward during your Move to Production passes. If you add new equations to your old release before doing your final Move to Production, you will need to add the same equation to your old Copy of Production (Source database, for your Move to Production pass) before your final Move to Production, in order to have it in your new release when you go live with the new release.

---

### Task 5-7-1: Running the Equation Conversion Reports

To run the equation conversion report:

1. Select Set Up HRMS, Upgrade, Reports, Equation Validation.
2. Select the Add a New Value tab.
3. Enter the run control ID *UPG\_EQTN*.
4. Click Add.
5. Click Run.
6. Select both reports to be run.
7. Click OK.
8. Click the Process Monitor link to monitor the program's process

The resulting report lists the equation name for those equations that need verification and a message about how the conversion program processed the equation. You will use this report to complete the next step.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

## Task 5-7-2: Reviewing Manual Conversion Candidates

In this step, you review the Manual Conversion Candidates report that you ran in the last step to verify that the equations converted correctly. If a converted equation is not correct, you will need to edit the converted equation.

To review manual conversion candidates:

1. Select Setup SACR, Common Definitions, Equation Engine, Equation Editor.
2. On the Find Existing Value tab, enter an equation name from the Manual Conversion Candidates report.
3. Click Search.
4. Review and edit the equation to make sure it will function correctly.
5. Set up global variables on the Equation Test Data page and test the equation using the TEST button on the Equation Engine edit page to ensure that the equation functions as expected.
6. Repeat steps 2 through 5 for the remaining equation names listed on the Manual Conversion Candidates report.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

## Task 5-7-3: Reviewing Equations

In this step, you verify that the equations listed in the report have been converted correctly. The following table shows the messages listed on the report and the action you need to take.

Message	Verification
Equation will not be converted because it is not in the conversion name table (18030, 75).	The conversion name from the report was not converted because it was never entered into the upgrade page filled out in task titled “Mapping Equation Values.” If you want to use this equation in the your new release, you will need to add the equation manually at this time.
Inserting missing ENDIF in equation, <i>Equation Name</i> line <i>line number</i> (14460, 405).	The end of the equation was reached and IF left open (no ENDIF was supplied). Review the equation to make sure that the if statements are completed correctly

Message	Verification
Dual sourced field <i>fieldname</i> has no associated equation in equation <i>Equation Name</i> (14460, 407).	This field can have two sources, and the associated equation needed to determine which equation to use was blank.
Interpreting assignment as equality test in <i>Equation Name</i> line <i>line number</i> (14460, 403).	An assignment operator was found within a condition statement. This will be interpreted as an equality test operator instead.
Warning: Incomplete conversion of IF in equation <i>Equation Name</i> line <i>line number</i> (14460, 402).	Within the equation there was an IF left open (no ENDIF was supplied). Review the equation to make sure that the if statements are completed correctly.
Ignoring unexpected THEN in equation <i>Equation Name</i> near line <i>line number</i> (14460, 401).	There is an extra THEN statement inside of an IF in the equation listed. Review the equation to make sure that the if statements are correct
Could not determine FIND NEXT table name in equation <i>Equation Name</i> (14460, 406).	A FIND NEXT statement was encountered, but the conversion program could not determine which table is referenced as having performed the FIND FIRST.
Interpreting equality test as assignment operator in <i>Equation Name</i> line <i>line number</i> (14460, 404).	An equality test operator was found where an assignment operator was expected. Interpreting this as an assignment operator.

To review equations:

1. Select Setup SACR, Common Definitions, Equation Engine, Equation Editor.
2. On the Find Existing Value tab, enter the equation name.
3. Click Search.
4. Review and edit the equation to make sure it will function correctly.
5. Set up global variables on the Equation Test Data page and test the equation via the TEST button on the Equation Engine edit page to ensure that the equation functions as expected.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

## Task 5-7-4: Reviewing Equation Security

During the upgrade conversion of equations, newly converted equations are assigned the security class PUBLIC, by default. You need to use Tree Manager to modify the security tree structure for equations to suit your business.

To modify the security tree structure for equations:

1. Select Tree Manager, Tree Manager.

2. Search for tree names beginning with *EQTN*.
3. Select EQTN\_IDAUTH\_TREE.
4. Add the authorizations for the converted equations to the appropriate node names within that tree structure and remove them from the PUBLIC node, as appropriate.

For example, converted Financial Aid equations might go into a new node that you create and call Financial Aid Administrators.

For details about how to maintain your Equation Security Trees, see the Equation Engine Programmer's Guide.

See *PeopleSoft Enterprise Campus Solutions 9.0 Application Fundamentals PeopleBook* "Equation Engine Programmer's Guide."

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Student Administration	All	All

---

## Task 5-8: Running the ECPrep Process

In this task you run the ECPrep process to update the ECOUTMAP.SQC file with the outbound EDI maps used by the Financial Aid module.

To run the ECPrep process:

1. Sign on to the Copy of Production database.
2. Select PeopleTools, EDI Manager, Monitor EDI Processing, Prepare Outbound Driver.
3. Select Add a New Value.
4. Enter the run control *FA\_ECPREP*.
5. Set the target directory for the ECOUTMAP.SQC file.
6. Select the check boxes for the *C004P* and *A004P* transaction IDs and click Save.
7. Click the Run button.
8. Click the Process Monitor link to monitor the process.

Run Parameters

Run Control ID: FA\_ECPREP

[Report Manager](#)
[Process Monitor](#)
[Run](#)

ECOUTMAP.SQC Directory:

\rtas080\test\_data\

Map Selection Criteria

Trans ID:

Select

Unselect

Refresh

Find | View All

First

1-3 of 3

Last

Select	Transaction ID	Map ID	Description
<input type="checkbox"/>	EERE	BCWCB	BC WCB Transactions
<input checked="" type="checkbox"/>	C004P	CL_CHNG_V4	CommonLine 4 Change Send
<input checked="" type="checkbox"/>	A004P	CL_ORIG_V4	CommonLine Application

ECOUTMAP.SQC Directory page

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	All	All

## Task 5-9: Running the Person Organization Audits

This section discusses:

- Understanding the Person Organization Audits
- Running Conversion Audits

### Understanding the Person Organization Audits

Run the conversion audits for records PER\_ORG\_INST and PER\_ORG\_ASGN - UVHCW01.

At the beginning of the upgrade you ran the Person Relationship Audit Report to list orphaned rows. If you did not clean up the orphans listed on those reports you will have orphans in the PER\_ORG\_INST and PER\_ORG\_ASGN. This report audits the load of record PER\_ORG\_ASGN against records PERSON and EMPLOYMENT. You will need to work with your Database Administrator to correct these orphans using your platform-specific SQL tool. These rows will not be accessible through the new PeopleSoft release.

This report will list the following:

- The audit of the load of PER\_ORG\_ASGN against PERSON lists EMPLIDs that are not in PER\_ORG\_ASGN.

Conversion of a commercial database should result in at least one row in PER\_ORG\_ASGN for every EMPLID.

- The audit of the load of PER\_ORG\_ASGN against EMPLOYMENT checks that every EMPLID/EMPL\_RCD was converted to PER\_ORG\_ASGN.

There should be a one-to-one relationship between the rows in EMPLOYMENT and the rows in PER\_ORG\_ASGN.

- Check that the number of rows inserted in record PER\_ORG\_ASGN is correct against PERSON and EMPLOYMENT.

## Task 5-9-1: Running Conversion Audits

Follow the steps below to run Conversion Audits for PER\_ORG\_INST and PER\_ORG\_ASGN.

To run the report:

1. Launch PeopleTools and sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, PER\_ORG Records Audits.
3. Click Add.
4. Enter the run control *UPG\_PER\_ORG*.
5. Click Run.
6. Click the Process Monitor link to monitor the program's process.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

## Task 5-10: Setting Up Pension

In the new PeopleSoft release, the system-delivered and your own customized database aliases that reference PERSONAL\_DATA will be converted to map to the records PERSON, PERS\_DATA\_EFFDT, ORIG\_HIR\_PER\_VW, or PERSON\_NAME based on the field name mapping. An audit report in the format, PAAUDALS\_yyyymmdd\_hhmm.LIS, will be produced during data conversion. This report lists the database aliases that could not be successfully converted. These reported exceptions will need to be fixed manually since they do not map to records that have an appropriate database alias key structure.

Three new database aliases are delivered as system data:

Alias Name	Record Mapping	Field Mapping
COMPANY	JOB	COMPANY

Alias Name	Record Mapping	Field Mapping
REG_REGION	JOB	REG_REGION
DC_FRACTN	PA_DC_FRACT	DC_FRACTION

During data conversion, any customer database alias names that match the above names will be renamed to preserve the mapping. Any conflicts will be reported in the audit file PCONVALS\_YYYYMMDD\_HHMM.LIS. You will need to analyze these conflicts to determine the impact on Pension plan implementation steps. You can use the Pension Alias Cross Reference utility to find all impacted items.

---

**Note.** Data conversion will not produce a report if there are no conflicts.

---

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Pension Administration	All	All

---

## Task 5-11: Reviewing the Benefits Limit Table

To simplify the maintenance of regulatory limits, and make their application more consistent, the Benefit Program has been removed as a key to the Limit Table. Maintaining virtually identical copies of each limit for each Benefit Program created significant redundant entry. You now only have to maintain a single version of each regulatory limit. During data conversion, the system consolidates all versions of existing limits into one “master” limit, combining them by Limit Type and Effective Date. This “master” limit contains comprehensive lists of deductions Subject to the Limit or Adjusting the Special Accumulator.

It is imperative that you carefully review the results of this consolidation, since a simple “mechanical” process is performed that could possibly not be consistent with your business policy. PeopleSoft suggests running the Limit Table report after upgrade is complete to assist in the verification of the new consolidated limits. Select Set Up HRMS, Product Related, Base Benefits, Rate/Rule Reports, Limit Table (BEN721).

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-12: Setting Up Stock Administration

This section discusses:

- Understanding Stock Administration



- Setting Up the Stock Table
- Setting Up the Stock Purchase Plan Rules

## Understanding Stock Administration

In this task, you set up Stock Administration for FAS 123 and APB 25.

In the new PeopleSoft release, new fields were added to the Stock Table to support FAS 123 and APB 25. The new fields added to the Stock Table are Original Fair Value Date, Accounting Standard, and FAS 123 Adoption Date. Manual changes are required to set the Original Fair Value Date on the Stock Table. The Original Fair Value Date should be populated with the fiscal year date beginning on or after 12/15/94 or 12/15/95. During the data conversion task, the Accounting Standard field defaults to *FAS 123*, and the FAS 123 Adoption Date field is then required. You will need to make manual changes if the Accounting Standard is APB 25, and the FAS 123 Adoption Date field is not required in this case.

New fields were also added to the Stock Purchase Plan Table to support FAS 123. The new fields added to the Stock Purchase Plan Table are FAS 123 Compensatory Plan Indicator and Valuation Plan Type. During the data conversion task, the FAS 123 Compensatory Plan Indicator field defaults to *Y*, unless the Purchase Price Rule is *Purchase FMV* and the Discount Percentage is less than or equal to 5%. Also during the data conversion task, the Valuation Plan Type field is set to *A* when the Purchase Price Rule is *Lower of Grant or Purchase FMV*. Changes will need to be made manually to set the Valuation Plan Type for all Stock Purchase Plans where the Purchase Price Rule is not equal to *Lower of Grant or Purchase FMV*.

### Task 5-12-1: Setting Up the Stock Table

Follow the steps below to set up the Stock Table.

To set up the Stock Table:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Stock, Stock Rules, Stock Table.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

### Task 5-12-2: Setting Up the Stock Purchase Plan Rules

Follow the steps below to set up the stock purchase plan rules.

To set up the stock purchase plan rules:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Stock, Purchase Plans, Stock Purchase Plan Rules.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-13: Validating Alternative Overtime Upgrade

If you requested the Alternative Overtime upgrade during the Making Functional Decisions task at the beginning of the upgrade, the upgrade process has populated the new ALT\_OT\_STATES table with data representing the U.S. states that require Alternative Overtime processing. This data is keyed by setID and EFFDT. For TableSet Control purposes, the ALT\_OT\_STATES table is in the new PY\_04 Record Group. The upgrade process created the new record group within every tableset, and assigned to it the default setID that is associated with that tableset.

Navigate to PeopleTools, Utilities, Administration, TableSet Control and review the default setID values that have been assigned to the PY\_04 Record Group, and ensure that they are appropriate to your requirements. The Alternative Overtime payroll processes will use COMPANY as the Set Control value for TableSet Controls when looking up the Alternative Overtime states. If all your companies will require Alternative Overtime processing for the same states, then you should only have data for one setID on the ALT\_OT\_STATES table. This setID should be assigned to the PY\_04 record group for all set control values. In this context, Set Control value = Company. Conversely, if the states will be different for some companies, then there should be data for separate setIDs on the ALT\_OT\_STATES table, and you should assign different setIDs to the PY\_04 record group.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

---

## Task 5-14: Updating Payroll Interface Definitions

This section discusses:

- Understanding Updates to Payroll Interface Definitions
- Running Validate PI Field References Report
- Updating PS Table Definitions
- Updating Field Definition Table
- Updating Instance Table Definitions

## Understanding Updates to Payroll Interface Definitions

In this task you perform steps to update the PeopleSoft Payroll Interface Definitions.

---

**Note.** Perform this task only if you have PeopleSoft Payroll Interface and currently have field entries that use record (table) names impacted by structural changes in the new PeopleSoft release.

---

A number of enhancements were made in the new PeopleSoft release that involved record structure changes that may impact Payroll Interface. Some of your existing Payroll Interface definitions may not be valid as a result of these changes. You must review your Payroll Interface definitions and update PS Table, Field Definition Table, and Instance Table, if needed.

Before updating Payroll Interface definitions, you need to do an analysis of structural changes in the new PeopleSoft release, to properly locate all impacted records and fields. For example, for personal data changes, you can review the appendix “Understanding Person Model Changes” and decide how you will update your tables.

### Task 5-14-1: Running Validate PI Field References Report

In this step, you run a report that lists all invalid record and field references as a result of new and modified record structures.

To run the Validate PI Field References report:

1. From your browser, sign on to your Copy of Production database, keeping the default browser address.
2. Select Set Up HRMS, Upgrade, Reports, Validate PI Field References.
3. Click Add a New Value.
4. Enter the run control ID *UPG\_PI\_DEFN*.
5. Click Add.
6. From the Check PI Definitions page, click Run.
7. Analyze the data presented in the report to decide whether any changes to Payroll Interface definitions are needed.

The report has three different sections:

- Invalid references in PS Tables
- PI Field Tbl
- PI Instance Tbl

If the report lists any data in any section, proceed with updating the corresponding definitions in the following steps.

#### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll Interface	All	All

## Task 5-14-2: Updating PS Table Definitions

In this step, you update any invalid references in PS Tables.

To update PS Table Definitions:

1. From your browser, sign on to your Copy of Production database, keeping the default browser address.
2. Select Set Up HRMS, Product Related, Payroll Interface, Interface Controls, PS Tables, Find an Existing Value.
3. Enter a Payroll Interface system ID listed in the PS Tables section of the Validate PI Field References report, if you see any.
4. Click Search.
5. Select Record (Table) Name listed in the PS tables section of the Validate PI Field References report.
6. On the PS Tables page, review and update the Field Details area.
7. Delete the fields from the Field Details of the records where they used to be.  
Then add the fields to the records where they were moved in this release.
8. Repeat steps 2 through 7 for each record listed in the PS Tables section of the report.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll Interface	All	All

## Task 5-14-3: Updating Field Definition Table

In this step, you update any invalid references in the PI Field Definition Table.

To update the Field Definition Table:

1. Select Set Up HRMS, Product Related, Payroll Interface, Interface Controls, Field Definition Table, Find an Existing Value.
2. Enter a Payroll Interface system ID, Payroll Interface field ID, and process type for fields listed in the PI Field Tbl section of the Validate PI Field References report, if you see any.
3. Click Search.
4. On the Interface Field2 page, review and update the field definition.
5. Repeat steps 1 through 4 for each field listed in the PI Field Tbl section of the report.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll Interface	All	All

## Task 5-14-4: Updating Instance Table Definitions

In this step, you update any invalid references in the PI Instance Table.

To update Instance Table definitions:

1. Select Set Up HRMS, Product Related, Payroll Interface, Interface Controls, Instance Table, Find an Existing Value.
2. Enter a Payroll Interface system ID and instance ID listed in the PI Instance Tbl section of the Validate PI Field References report, if you see any.
3. Click Search.
4. Review and update the PeopleSoft Record Name and PeopleSoft Field1 Name fields.
5. Repeat steps 1 through 4 for each instance ID listed in the PI Instance Tbl section of the report.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll Interface	All	All

---

## Task 5-15: Upgrading Rules

This section discusses:

- Understanding Rules Upgrade
- Running the Dropped Temp Tables Report
- Recompiling Template Built Rules
- Customizing Template Built Rules
- Reviewing and Modifying User Exit Rules
- Modifying SQL Statements for PROCESS\_INSTANCE

### Understanding Rules Upgrade

This task applies only if you are upgrading the PeopleSoft HRMS Time and Labor product. Many SQL objects may have been modified for performance and functionality enhancements.

### Task 5-15-1: Running the Dropped Temp Tables Report

In this step, you run a report to list any temporary tables that you use to customize rules that were dropped for the new release. You will fix these rules as you complete the steps in this task. Refer to this report to see which rules have temporary tables to be fixed. After you complete this step, rerun this report to validate that all rules were upgraded.

The script is:

```
UVTLW10.SQR
```

To run the Deleted Temp Table Report:

1. Select Set Up HRMS, Upgrade, Reports, TL Temp Tables List.
2. Click Add a New Value.
3. Enter the run control ID *UPG\_RULE*.
4. Click Add.
5. Click Run.
6. Click the Process Monitor link to monitor the Program's process.

The resulting report lists any of the rules that use temp tables that were dropped in the new release. Review the rules and change them to work in the new system.

Check to see whether the temp table mentioned in the report exists in the database. If it exists, then add that table to the properties for the TL\_TIMEADMIN application engine. If the temp table does not exist, then check the object where that table is used and fix the object.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All

## Task 5-15-2: Recompiling Template Built Rules

All rules that you have built from rule templates need to be recompiled. To recompile Template-Built Rules, you will need to follow the steps below.

To recompile Template-Built Rules:

1. If you are using Referential Integrity, export any data you have in TL\_RI\_CNTRL\_TBL before you clear the Production Environment check box option on the Time and Labor Installation page.  
This table will be deleted once you clear the check box, and you will re-import this data later in the procedure.
2. Select Set Up HRMS, Install, Product and Country Specific, Time and Labor Installation.
3. Clear the Production Environment check box, as shown below, and click Save.

**Time and Labor Installation**

**System Options**

☐ Production Environment ☒ Interface with PS/EPM ☒ Interface with PS/Projects

☐ Referential Integrity ☒ Publish unmapped TRC as No Pay ☒ Calculate Estimated Gross

**TL/NA Payroll Options**

Number of Months to Process: 24 ☒ Process Prior Paygroup Time with Current Paygroup

Concurrent Load Time and Labor Cleanup: 1 ☒ Concurrent Load Time and Labor Runs

☒ Set Ignore Status for Not Ok to Pay

**Validation Options**

\*Leave Balances: On-line and Batch Validation Set: ST\_ALL

\*Compensatory Balances: On-line and Batch Chartfields Validation: None

**Time Administration Options**

☒ Automatic Time Admin Run ☒ Continue with Exceptions

Max Employees in a Batch: 10000 ☒ Include Multiple Jobs

**On-line Rules Options**

☒ Run On-line Rules Maximum On-line Rules: 25

**TCD File Integration**

Outbound File Directory: c:\temp\

Time and Labor Installation page

4. Select Set Up HRMS, Product Related, Time and Labor, Rules and Workgroups, Template-Built Rules.
5. Instead of entering any values in any field, click Search to display all the Template-Built Rules in your system.

**Template-Built Rules**

Enter any information you have and click Search. Leave fields blank for a list of all values.

**Find an Existing Value** **Add a New Value**

Template Type: =

Rule Template: begins with

Rule ID: begins with

**Search** **Clear** [Basic Search](#) [Save Search Criteria](#)

**Search Results**

View All First 1-100 of 109 Last

Rule ID	Description
GXAMDFLT	Absence Management Default
GXTLDFLTRC	GXTLPRG DEFAULT TRC
K0MDFLT100	Default TRC 100
K0MDFLTRGR	Default TRC K0RGR
K0WDFLT	Absence Management Default
KABONAPP	Bonus Approval
KACALLOUT	Guaranteed 4 Hours Pay
KACALLOUT1	Callout Pay on Unscheduled Day
KADFLTRC	Default TRC (Meal & Break)
KADFTRC1	Default TRC (Break)
KADFTRC2	Default Salary TRC KARSL
KA0VT1.5	Daily Time and Half Overtime
KA0VT2.0	Pay Double Time After 1.0 Hours

Template-Built Rules page

6. Select each rule to review it, making changes if required, then click Save.

Saving the component will recompile the rule for you. If no changes are required, you will need to make some minor change to trigger the save. It is recommended that you change the Short Description field by either changing the current text or copying the AE Section field and pasting it into the Short Description field.

**Note.** A new field, Create/Replace, was added to all of the Holiday Template-Built Rules. This field could have a value of either *create* or *replace*. This gives you the opportunity to either create new payable time for a holiday worked or replace the time reported for the holiday. If you have not applied the fix or the bundle to add this field, the field may not be populated. If you do not select a value for this field, your holiday rules may not work properly in the new PeopleSoft release. Therefore, for all of your Holiday Template-Built Rules, make sure you select a value for the Create/Replace field on the second page and save the rule. Saving the rule will recompile and update the Application Engine code with the proper SQLs.

The screenshot displays the 'Template-Built Rules' page in the 'Holiday' tab. The left-hand navigation pane shows a tree structure with 'Template-Built Rules' highlighted. The main content area is titled 'Template Header' and includes the following information:

- Rule Template:** TEMPLATE060
- Rule ID:** KUHOLIDAY
- Paid Holiday Rule**

Below the header is an 'Explanation' section containing the following text:

Pay the following Time Reporters (S) 8 hours of TRC KUHOL for holiday pay.

This template can be used to pay Holiday Pay to all Time Reporters.

The specified Holiday TRC will be created in addition to reported/scheduled TRC.

Buttons for 'Update Rule Text' and 'Modify Conditions' are located to the right of the explanation text.

The 'Criteria (Conditions)' section shows:

- Empl Type List:** TL\_SALARIED (with a 'View List' button)

The 'Result (Action)' section shows:

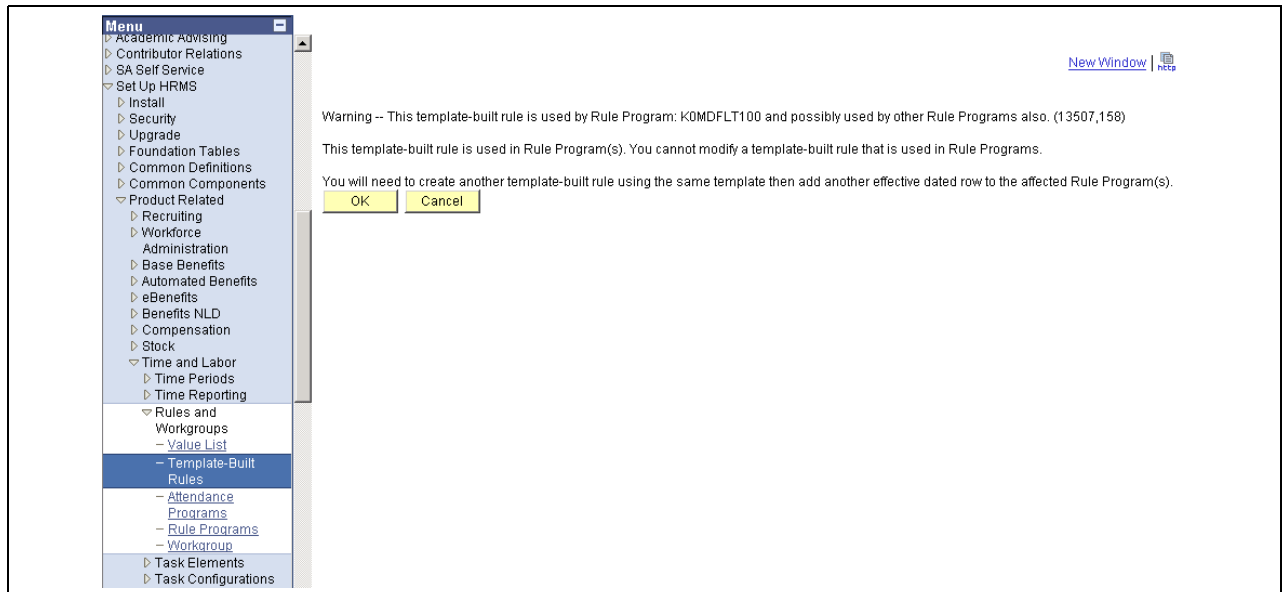
- Quantity:** 8.000000
- Create/Replace:** Create (this dropdown menu is circled in red)

At the bottom of the page are several action buttons: 'Save', 'Return to Search', 'Next in List', 'Previous in List', 'Notify', and 'Add'.

Template-Built Rules page; Holiday tab

If the rule being recompiled is included in rule programs, a warning message similar to this one will be displayed:





Template-Built Rules - warning page

- After you have finished recompiling all of your Template-Built Rules, select the Production Environment check box option on the Time and Labor Installation page to enable the Referential Integrity process.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All

## Task 5-15-3: Customizing Template Built Rules

This section discusses:

- Understanding Template-Built Rules Customizations
- Customized Template-Built Rules
- Rules Created from Actions and Conditions or SQL Objects

### Understanding Template-Built Rules Customizations

In this step you customize your Template-Built Rules as well as review and make changes to rules created from Actions and Conditions or SQL objects.

### Customized Template-Built Rules

Customized Template-built Rules are rules that you have created using delivered-rule templates and have modified to meet your specific needs. To upgrade these rules, you first need to determine the effort required to modify the current rules (created in PeopleSoft 8.0 SP1) versus the effort required to create new rules from delivered-rule templates (in PeopleSoft 9.0) to be modified.

If you think that it will be faster to create a new rule from a rule template than to modify that rule, you will need to apply the same changes that you applied to PeopleSoft 8.0 SP1 Template-Built Rules.

If you choose to modify the current rule (created in PeopleSoft 8.01), you will need to review and modify all the steps in the rule to accommodate the addition of the PROCESS\_INSTANCE field to temp tables used in the rule.

If you have created your own temporary tables to be used in custom rules, you will need to add PROCESS\_INSTANCE as the first key field and modify your SQL statements accordingly.

Changes to SQL statements have to be made on the Define Actions, Define Conditions, or SQL Objects pages. To review custom rules, select Set Up HRMS, System Administration, Utilities, Build Time and Labor Rules, Rules, and enter the first few characters of the rule ID that will follow your naming convention for custom rules, as shown in the example below:

**Rules**

Enter any information you have and click Search. Leave fields blank for a list of all values.

[Find an Existing Value](#) [Add a New Value](#)

Rule ID:

Description:

☐ Case Sensitive

[Search](#) [Clear](#) [Basic Search](#) [Save Search Criteria](#)

**Search Results**

[View All](#) First  [Last](#)

Rule ID	Description
GXAMDFLT	Absence Management Default
GXTLDFLTRC	GXTLPRG DEFAULT TRC
K0MDFLT100	Default TRC 100
K0MDFLTRGR	Default TRC K0RGR
K0WDFLT	Absence Management Default
KABONAPP	Bonus Approval
KACALLOUT	Guaranteed 4 Hours Pay
KACALLOUT1	Callout Pay on Unscheduled Day
KADFLTRC	Default TRC (Meal & Break)
KADFTRC1	Default TRC (Break)
KADFTRC2	Default Salary TRC KARS
KAOVT1.5	Daily Time and Half Overtime

Rules page

## Rules Created from Actions and Conditions or SQL Objects

For rules created from Actions and Conditions or SQL objects, you need to review and make necessary changes as mentioned in the Customized Template-Built Rules section above. If you used delivered rules objects (Actions and Condition, SQL Objects, or Temp Tables) in your custom rules without any modifications, you only need to recompile them.

To recompile custom Template-Built rules:

1. Select Set Up HRMS, System Administration, Utilities, Build Time and Labor Rules, Rules.
2. On the Define Rule Header tab, click the Compile Rule button.

The screenshot shows the 'Define Rule Header' page. On the left is a menu tree with 'Rules' selected. The main area is titled 'Rule Header Definition' and contains the following fields:

- Rule ID:** GXTLDFLTRC
- Description:** GXTLPRG DEFAULT TRC
- Short Description:** GXTLPRG D
- Time Period ID:** PSDAY
- AE Section:** GXTLDFLT
- Run Location:** Batch Only
- Compiled?:** Y
- Last Compiled:** 07/05/2006 11:29:58.823000AM
- Explanation:** This rule will assign any hours without a Time Reporting Code (TRC) to TRC GXRWK.

Buttons for 'Compile Rule' and 'Rule SQL Text' are located at the top right of the main pane.

Define Rule Header page

If you have cloned delivered rules objects and modified them to be used in your custom rules, you need to review and modify these custom rules. You will need to either make changes directly in the current rules objects (cloned and modified in your previous PeopleSoft release) or clone PeopleSoft 9.0-delivered rules objects to modify. Then you will need to recompile these rules.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All

## Task 5-15-4: Reviewing and Modifying User Exit Rules

If you have created any rules as User-Exits, you need to review and modify them directly in the rules Application Engine library, TL\_TA\_RULES.

To review and modify User-Exit rules:

1. Launch Application Designer.
2. Open the Application Engine program TL\_TA\_RULES.
3. On the Application Engine Definition page, look for the Application Engine section.
4. Modify your SQL steps, as appropriate, to accommodate PROCESS\_INSTANCE and other changes.
5. Click Save to save the Application Engine section.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All

## Task 5-15-5: Modifying SQL Statements for PROCESS\_INSTANCE

In the new release, many SQLs were modified by adding the PROCESS\_INSTANCE field. Any custom SQLs that you have created will need to be modified to incorporate the PROCESS\_INSTANCE field.

The example below will help you understand how you should make these changes.

The SQL statement in 8.01 is as follows:

```
UPDATE %Table(TL_IPT2)
SET RULE_FLAG = (SELECT MAX(A.SEQ_NBR)
FROM %Table(TL_IPT1) A
WHERE A.EMPLID = %Table(TL_IPT2).EMPLID
AND A.EMPL_RCD = %Table(TL_IPT2).EMPL_RCD
AND A.DUR = %Table(TL_IPT2).DUR)
WHERE EXISTS (SELECT 'X'
FROM %Table(TL_IPT1) A1
WHERE A1.EMPLID = %Table(TL_IPT2).EMPLID
AND A1.EMPL_RCD = %Table(TL_IPT2).EMPL_RCD
AND A1.DUR = %Table(TL_IPT2).DUR)
```

You should modify the above SQL statement in the new release as shown (changes are in bold) :

```
UPDATE %Table(TL_IPT2)
SET RULE_FLAG = (SELECT MAX(A.SEQ_NBR)
FROM %Table(TL_IPT1) A
WHERE A.PROCESS_INSTANCE = %Table(TL_IPT2).PROCESS_INSTANCE
AND A.EMPLID = %Table(TL_IPT2).EMPLID
AND A.EMPL_RCD = %Table(TL_IPT2).EMPL_RCD
AND A.DUR = %Table(TL_IPT2).DUR)
WHERE PROCESS_INSTANCE = %Bind(PROCESS_INSTANCE)
AND EXISTS (SELECT 'X'
FROM %Table(TL_IPT1) A1
WHERE A1.PROCESS_INSTANCE = %Table(TL_IPT2).PROCESS_INSTANCE
AND A1.EMPLID = %Table(TL_IPT2).EMPLID
AND A1.EMPL_RCD = %Table(TL_IPT2).EMPL_RCD
AND A1.DUR = %Table(TL_IPT2).DUR)
```

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Time and Labor	All	All

---

## Task 5-16: Deleting Resume Files

Perform this task only if you have licensed eRecruit Resume Processing and have resumes on a file system.

The resume file name in the Resume Text Table was updated to reflect the full resume file name on your file system during data conversion. If you specified HRS\_ATTACHMENTS as the record URL during the task Making Functional Decisions, your resumes migrated from the file server into the database. If you are continuing to use a file system, your resume files moved to the alternate location specified in this URL field. You need to manually delete the resume files from the old location once the rename and migrate processes have completed successfully.

See "Prepare Your Database," Making Functional Decisions.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-17: Setting Up General Ledger Interface

This section discusses:

- Understanding GL Interface Setup
- Setting Up the Liability Accounts
- Setting Up the GL Activity Groupings
- Setting Up the Expenses ChartField Mappings

## Understanding GL Interface Setup

In this task you will set up the tables that are used for GL Interface based on the decisions you made after reviewing audit reports and discussing ChartFields combinations with General Ledger staff.

---

**Note.** Run this task only if you are using non-commitment accounting in Payroll for North America.

---

---

**Important!** Once you create your copy of the production database in the initial pass, you should not make any changes in your production database for Company, Company State Taxes, Company Local Taxes, Wage Loss Plan Table, and Deduction Table until you complete the upgrade. If you make any changes to any of the tables listed above, you will not be able to use import scripts during the Move to Production passes to import data that you have set up in the initial pass. You will either have to synchronize the data between production and the copy of the production you created in the initial pass before you set up the data and do the export, or manually set up data in your new copy of the production during the Move to Production pass.

---

See Appendix: “Understanding Dynamic ChartFields Changes.”

See “Prepare Your Database,” Running Application Audits, Running Dynamic ChartFields Audits.

## Task 5-17-1: Setting Up the Liability Accounts

This section discusses:

- Understanding Liability Accounts
- Setting Up NetPay, Direct Deposit, and Federal Tax Liability Accounts
- Setting Up State Tax Liability Accounts
- Setting Up Local Tax Liability Accounts
- Setting Up Canadian Tax Liability Accounts
- Setting Up Deduction Liability Accounts

### Understanding Liability Accounts

Use the data listed in the Liability ChartField Usage audit report to work with your General Ledger staff and make decisions about how you want to set up the liability accounts that will offset the credit transactions for payroll activities. Your setup will vary depending on whether you are using US Payroll or Canadian Payroll. Follow the steps in this section to set up the liability accounts that are applicable to the payroll type you are using.

---

**Note.** NetPay, Direct Deposit, and Deduction liability accounts are for both US Payroll and Canadian Payroll. Federal Tax, State Tax, and Local Tax liability accounts are for US Payroll only. Canadian Tax liability accounts are for Canadian Payroll only.

---

### Setting Up NetPay, Direct Deposit, and Federal Tax Liability Accounts

To set up liability accounts for NetPay, Direct Deposit, and Federal Tax:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Foundation Tables, Organization, Company.
3. Enter the Company.
4. Click Search.
5. Select the Default Settings tab and the General Ledger Accounts link.
6. Enter the liability accounts and click OK.
7. Click Save.
8. Repeat steps 1 through 7 for each company that needs to be set up.

## Setting Up State Tax Liability Accounts

To set up liability accounts for State Tax:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Federal/ State Taxes, Company State Tax Table.
3. Enter the Company/State combination.
4. Click Search.
5. Select the GL Accounts tab.
6. Enter the liability accounts.
7. Click Save.
8. Repeat steps 1 through 7 for each Company or State combination that needs to be set up.

## Setting Up Local Tax Liability Accounts

To set up liability accounts for Local Tax:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Local Taxes, Company Local Tax Table.
3. Enter the Company/State/Locality combination.
4. Click Search.
5. Enter the liability accounts.
6. Click Save.
7. Repeat steps 1 through 6 for each Company/State/Locality combination that needs to be set up.

## Setting Up Canadian Tax Liability Accounts

To set up liability accounts for Canadian Tax:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Canadian Taxes, Wage Loss Plan Table.
3. Enter the Company/Wage Loss Replacement Plan combination.
4. Click Search.
5. Enter the liability accounts.
6. Click Save.
7. Repeat steps 1 through 6 for each Company or Wage Loss Replacement Plan combination that needs to be set up.

## Setting Up Deduction Liability Accounts

To set up liability accounts for Deductions:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Deductions, Deduction Table.
3. Enter the Plan Type/Deduction Code combination.

4. Click Search.
5. Select the Process tab.
6. Enter the liability accounts.
7. Click Save.
8. Repeat steps 1 through 7 for each Plan Type or Deduction Code combination that needs to be set up.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

## Task 5-17-2: Setting Up the GL Activity Groupings

Use the data listed in the Chartkey Usage by Expense Types audit report to work with general ledger staff and make decisions about how you want to set up the payroll activities. The payroll activities are mapped to ChartFields combinations later in the upgrade.

Follow the steps below to set up the GL Activity Groupings.

To set up the GL activity groupings:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, GL Interface, GL Activity Grouping.
3. Click Add a New Value.
4. Enter Company/GL Group Name.
5. Click Add.
6. Enter data on the Earnings Expenses tab, if applicable.
7. Enter data on the Deduction Expenses tab, if applicable.
8. Enter data on the Tax Expenses tab, if applicable.
9. Click Save.
10. Repeat steps 1 through 9 for each Company/GL Group Name you need to set up.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

## Task 5-17-3: Setting Up the Expenses ChartField Mappings

Use the data listed in the Chartkey Usage by Expense Types and Chartkey Usage by Company and Business Unit audit reports to work with General Ledger staff and make decisions about how you want to map payroll activities to a ChartFields combination.



To set up the Expenses ChartField Mappings:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, GL Interface, ChartField Expense Mapping <country> where <country> is, for example, CAN or USA.
3. Click Add a New Value.
4. Enter Company, Business Unit, and Mapping Level.  
Depending on the mapping level you selected, enter additional information if needed.
5. Click Add.
6. Enter data on the Earnings Mapping tab, if applicable.
7. Enter data on the Deductions Mapping tab, if applicable.
8. Enter data on the U.S. Tax Mapping tab, if applicable.
9. Enter data on the Canadian Tax Mapping tab, if applicable.
10. Click Save.
11. Repeat steps 1 through 10 for each Company, Business Unit, and Mapping Level you need to set up.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	Payroll for North America	All	All

---

## Task 5-18: Validating EE Garn Payee Data

This section discusses:

- Understanding EE Garn Payee Data Validation
- Running the EE Garn Payee Data Validation Report
- Updating Garn Payee Data

### Understanding EE Garn Payee Data Validation

In this task you perform steps to update payee data on garnishment orders, if needed.

---

**Note.** Run this task only if you are using AP Interface and run the Exporting Non-Tax Deductions program.

---

See Appendix: “Understanding Garnishments Changes.”

### Task 5-18-1: Running the EE Garn Payee Data Validation Report

In this step, you run a report that lists garnishment orders whose payee has either different invoicing and remitting Vendor ID or multiple locations.

To run the EE Garn Payee Data Validation Report:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, EE Garn Payee Data Validation.
3. Click Add a New Value.
4. Enter the run control ID *UPG\_GARN\_PAYEE*.
5. Click Add.
6. On the EE Garn Payee Data page, click Run.
7. Analyze the data presented in the report to decide if any input is needed.

The report has four different sections:

- Active orders with different invoicing and remitting Vendor ID.
- Inactive orders with different invoicing and remitting Vendor ID.
- Active orders with multiple locations.
- Inactive orders with multiple locations.

If there is any active order listed in the report, you are expected to review it and decide if you need to make any changes or not. The active orders have the status Approved or Received and they will continue to be processed after the upgrade. If the payee data is not correct, the Extract Non-Tax Deduction program will not create the invoice correctly or it will not create the invoice at all, depending on your setup.

This report will insert all active orders listed in the report into a new record that will be accessible in the online component. The next step provides instructions for accessing the online component and making the updates.

If there is any inactive order listed in the report, it is up to you to decide whether you want to make any updates to history data or not. Because these orders do not have Approved or Received status, they will not be processed in subsequent pay calculations and leaving them as is will not have any impact on creating the invoices.

The inactive orders will not be automatically inserted and available in the online component. If you decide you want to make the updates to history data, you can access the online component and manually add the orders you want to update. The next step provides instructions for accessing the online component and adding the history records.

This report will run in each upgrade pass, but it will only list the data that you need to review; for example, the initial pass listed 10 active orders, which you have reviewed and updated. The Move to Production passes will list additional orders only if you have created a new order between two passes that matches the report selection criteria; otherwise, it will be empty.

---

**Note.** If the final Move to Production pass lists any data in this report and you decide you need updates, you will do the updates directly in Create Garnishments or Payee Table in your production database before you go live.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-18-2: Updating Garn Payee Data

This section discusses:

- Understanding Updating Garn Payee Data
- Updating Active Garnishment Orders
- Updating Garnishment Orders After Final Move to Production Pass
- Adding Inactive Garnishment Orders

### Understanding Updating Garn Payee Data

In this step, you update payee data on active garnishment orders, if needed, or add inactive garnishment orders that have payee data you want to update.

### Updating Active Garnishment Orders

To update active orders:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Define Defaults, Upgrade EE Garnishment Orders.
3. Select the Upgrade Garn Spec tab.
4. Review the data and update the Location and Remit Address fields, if needed.
5. Click Save.

There are several things to keep in mind when you are reviewing the data on this page:

- The orders are inserted here because they match certain criteria, but that does not mean the data really needs to be updated. For example, payee XYZ has two locations and two remit addresses. Order ABC has payee XYZ with location 1 and remit address 1, order DEF has payee XYZ with location 2 and remit address 2. Both orders will be listed on this page because their payee has multiple locations, but the location and remit address might be correct for each order. If that is true, you do not have to do anything; the conversion will use the prepopulated data to do updates in the next upgrade pass, but the data will not really change. If you think that order DEF should also have location 1 and/or remit address 1, you should make the update on this page and the conversion programs will use your input to update the data on the garnishment order in the next pass.
- You can only update the Location and Remit Address fields for the orders that are automatically inserted by the validation report onto this page.

You will get an error message if you try to update any other field in the row.

- You can delete the row on this page, but when you run the report the next time, the validation will find all orders that match the selection criteria and insert the same row again, if it is still applicable.

The same criteria will most likely still be applicable, unless you have changed the payee setup in the Payee Table between upgrade passes.

- The Used field on this page may have multiple values.

This field is display only because it keeps track of why data is inserted, whether data passed through conversion, and whether data updated the real garnishment order or not. Review the translate values for this field to see details about each value. The rows created by the first section of the report will have the value *D* in the Used field because they have a payee with different invoicing and remitting vendors. The rows created by the third section of the report will have the value *L* in the Used field because they have a payee with multiple locations.

- If the payee has setup that matches both criteria, it will be listed and inserted into this page only once.  
You will have to ensure your input will correctly update data for both scenarios.
- In the following pass, the Used field will have a different value, depending on whether it actually updated the real order or not.

The conversion program will update the value of this field to *U* when the orders are updated, which will happen if the Location and Remit Address fields have any values. The conversion program will update the value of this field to *X* when the orders are not updated, which will happen if the Location or Remit Address fields do not have any values.

- The Used field will be very important after running the final upgrade pass.

If there is any row in the data with the value *D* or *L* in the Used field, you will have to access the garnishment order and do manual updates where applicable, because there will be no more upgrade conversion after the last validation. These entries would be created by the last validation report and most likely would be for the garnishment orders that were newly created in the Copy of Production since the last time you took the copy.

## Updating Garnishment Orders After Final Move to Production Pass

To manually update garnishment orders after the final Move to Production pass:

1. Sign on to the Copy of Production database.
2. Select Payroll for North America, Employee Pay Data <country>, Deductions, Create Garnishments where country is, for example, CAN, USA, or USF.
3. Enter the employee ID and company that needs manual updates.
4. Click Search.
5. If there is more than one garnishment order for the same employee, select the row with the garnishment ID that needs to be manually updated.
6. Select the Garnishment Spec Data 2 tab.
7. Update any of the following fields, as needed: SetID, Vendor ID, Location, Remit Address.
8. Click Save.

## Adding Inactive Garnishment Orders

To add inactive orders:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Define Defaults, Upgrade EE Garnishment Orders.
3. Select the Upgrade Garn Spec tab.
4. Click the “+” button to insert a new row.
5. Enter values for the fields EmplID, Company, GarnishmentID, SetID, and VendorID from the inactive orders sections of the report for each order you want to update.

Enter values for the Location and Remit Address fields from the available prompt lists, selecting the correct value for each order you are updating.

6. Click Save.

There are several things to keep in mind when you are adding new data on this page:

- All manually inserted rows will initially have the value *O* in the Used field, marking them as online entry.

- The system will not prevent you from entering any combination of data on this page.

It does not check whether you are entering only inactive orders. It is your responsibility to decide what updates you are making and add corresponding data on this page.

- In the following pass, the Used field will get a different value, depending on whether it actually updated the real order or not.

The conversion program will update the value of this field to *U* when the orders are updated, which will happen if the Location and Remit Address fields have any values. The conversion program will update the value of this field to *X* when the orders are not updated, which will happen if the Location or Remit Address fields do not have any values.

- The Used field will be very important after running the final upgrade pass.

If there is any row in the data with the value *O* in the Used field, you will have to access the garnishment order and do manual updates where applicable, because there will be no more upgrade conversion after the last validation. Follow the above instructions on how to access the garnishment order and fix the data manually.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-19: Setting Up US Custom Garnishment Rules

This section discusses:

- Understanding US Custom Garnishment Rules Setup
- Creating US Custom Garnishment Rules
- Cloning US Custom Garnishment Rules

### Understanding US Custom Garnishment Rules Setup

In this task you set up U.S. garnishment rules.

---

**Note.** Run this task only if you used to maintain any U.S. garnishment rules in prior releases, and you have decided that you will still need them after the upgrade.

---

In PeopleSoft 9, there are two ways to set up garnishment rules:

- Create the U.S. custom rules by entering the data in the online component.
- Clone one of the existing rules and save it as a new custom rule.

In an ideal situation, manual setup of custom rules should be done only during the Initial pass. However, it is possible due to timing of the upgrade passes, that you have added more custom rules since the last pass. You can compare the US Custom Garnishment Rules Audit report from the previous and the current pass to see whether any custom rules have been added since the previous pass. If there are some new rules, and you decide you need them in future, you can set them up during the Move To Production Pass, but this applies only to any new rules. The rules you have already set up in prior passes will be copied during the upgrade by running the export and import scripts for U.S. custom garnishment rules.

See Appendix: “Understanding Garnishments Changes.”

See “Prepare Your Database,” Running Application Audits, Running Upgrade Reports, Running US Custom Garnishment Rules Audit.

See Validating EE Garn Rule Data.

See “Apply Application Changes,” Completing Application Processes, Exporting US Custom Garnishment Rules and Importing US Custom Garnishment Rules.

See *PeopleSoft Enterprise Payroll for North America 9.0 PeopleBook*.

## Task 5-19-1: Creating US Custom Garnishment Rules

In this step you create new garnishment rule by using the online component to enter the required data.

To create a U.S. custom garnishment rule:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Garnishments, Rules Table.
3. Click Add a New Value.
4. Enter values in the Law Source and Rule ID fields for the rule that you want to create.
5. Click Add.
6. Enter all required data on each tab of the component.

You can use the U.S. Custom Garnishment Rule Audit report to find out what values you need to enter.

7. Click Save.
8. Repeat steps 3 through 7 for each U.S. garnishment rule that you need to set up.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-19-2: Cloning US Custom Garnishment Rules

In this step, you create a new garnishment rule by cloning an existing garnishment rule.

To clone any existing rule as new custom rule:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Product Related, Payroll for North America, Garnishments, Clone Rules.

3. Click Add a New Value.
4. Enter the run control ID *UPG\_CLONE*.
5. Click Add.
6. Enter values in the Law Source, RuleID, and Effective Date fields for the rule you want to clone.
7. Enter values in the Law Source, Rule ID, and Effective Date fields for the new rule you want to create.
8. If you want to create more than one rule, insert a new row in the New Garnishment Rule group box.
9. Repeat steps 7 and 8 for each rule you want to create.
10. Click Run.

There are several things to keep in mind when cloning existing rules:

- You can repeat the cloning process many times, using different rules to create as many new rules as you want.
- The new rules will be exactly the same as the rule used for cloning, except for the values in the Source Law, Rule ID, and Effective Date fields, which you entered on the run control page of the cloning process.
- You should try to find the rule that has the same formula as you need in your new rule.

If there is no rule with the identical formula, you can find a rule with the most similar formula and use it for cloning.

- After the rules are cloned, you can access the Garnishment Rules Tables and modify any of the cloned values.

For example, you can update the formula steps by deleting old or adding new steps, or update exemption variables or any other field that you want to change.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-20: Validating EE Garn Rule Data

This section discusses:

- Understanding EE Garn Rule Data Validation
- Running the EE Garn Rule Data Validation Report
- Updating Garn Rule Data

## Understanding EE Garn Rule Data Validation

In this task, you perform steps to review and update rule data on garnishments orders, if needed.

The purpose of this task is to find all orders that reference rules that do not exist in garnishment rules tables, because you might have some U.S. custom rules that you do not need. However, this task will not only check orders for U.S. custom rules, it will check orders for any rule that might not be defined in garnishment rules tables. We recommend that you run this task to check the validity of your data even if you do not have any U.S. custom rules.

See Appendix: “Understanding Garnishments Changes.”

See Setting Up US Custom Garnishment Rules.

See “Apply Application Changes,” Completing Application Processes, Exporting US Custom Garnishment Rules and Importing US Custom Garnishment Rules.

## Task 5-20-1: Running the EE Garn Rule Data Validation Report

In this step, you run a report that lists garnishment orders whose rules do not exist in Garnishment Rules Tables.

To run the EE Garn Rule Data Validation Report:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Reports, EE Garn Rule Data Validation.
3. Click Add a New Value.
4. Enter the run control ID *UPG\_GARN\_RULE*.
5. Click Add.
6. On the EE Garn Rule Data page, click Run.
7. Analyze the data presented in the report to decide how you will correct the data.

The report has two different sections:

- Active orders with rules missing in Garnishments Rules Tables.
- Inactive orders with rules missing in Garnishments Rules Tables.

After reviewing the report, you can decide how to fix the data depending on what type of rule is missing.

If the rule is a U.S. custom garnishment rule, you have the following two options:

- You can create the missing rule in Garnishments Rules Tables by following the instructions in Setting Up US Custom Garnishment Rules.
- You can update garnishments orders and replace the missing rule with the existing rule by following the instructions in the step below.

If the rule is a Canadian custom garnishment rule, you have following two options:

- You can create the missing rule in Garnishments Rules Tables CAN, which must be done in the Copy of Production before the tables are renamed.

This way the Canadian custom rules will be copied during the conversion from renamed tables into a new structure of garnishments rules tables.

- You can update garnishments orders and replace the missing rule with the existing rule by following the instructions in the step below.



If there is any active order listed in the report, it is expected that you will fix the data using one of the above options. The active orders have the status Approved or Received. They will continue to be processed after the upgrade. If the rule is not defined in Garnishment Rules Tables, the pay calculation program will issue an error for the employee.

This report will insert all active orders listed in the report into a new record that will be accessible in the online component. The next step provides instructions for accessing the online component and making the updates. If you are planning to create the missing rule, simply ignore the entries created by the report.

If there is any inactive order listed in the report, it is up to you to decide whether you want to make any updates to history data or not. Because these orders do not have Approved or Received status, they will not be processed in following pay calculations, and leaving them as is will not have any impact on payroll calculations.

The inactive orders will not be automatically inserted and available in the online component. If you decide that you want to make updates to history data, you can access the online component and manually add the orders you want to update. The next step provides instructions for how to access the online component and add the history records. If you decide to create the missing rule, do not add any history rows into the online component.

This report will run in each upgrade pass, but it will only list the data that you need to review; for example, the initial pass listed 10 active orders, which you have reviewed and updated. The Move to Production passes will list additional orders only if you have created a new order between two passes that matches the report selection criteria; otherwise, it will be empty.

---

**Note.** If the final Move to Production pass lists any data in this report and you decide you need updates, you will do the updates directly in Create Garnishments, Garnishments Rules Tables, or Garnishments Rules Tables CAN in your production database before you go live.

---

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-20-2: Updating Garn Rule Data

This section discusses:

- Understanding Updating Garn Rule Data
- Updating Active Garnishment Orders
- Updating Garnishment Orders After Final Move to Production Pass
- Adding Inactive Garnishment Orders

### Understanding Updating Garn Rule Data

In this step, you update Garn Rule Data on active garnishments orders, if needed, or add inactive garnishment orders that have Garn Rule data that you want to update.

### Updating Active Garnishment Orders

To update active orders:

1. Sign on to the Copy of Production database.

2. Select Set Up HRMS, Upgrade, Define Defaults, Upgrade EE Garnishment Orders.
3. Select the Upgrade Garn Rule tab.
4. Review the data and select values in the New Law Source and New Rule ID fields from the available prompt lists, selecting the correct value for each order you are updating.
5. Click Save.

There are several things to keep in mind when you are reviewing the data on this page:

- The orders are inserted here because they match certain criteria, but that does not mean you have to update the order and change the rule.

You can add the missing rule into Garnishment Rules Tables and ignore the entries listed in this page. If you do not provide new values for the law source and rule ID, the conversion will not make any updates in the next pass.

- You can only update the New Law Source and New Rule ID fields for the orders that are automatically inserted by the validation report onto this page.

You will get an error message if you are trying to update any other field in the row.

- You can delete the row in this page, but when you run the report the next time, the validation will find all orders that match the selection criteria and insert the same row again, if it is still applicable.

The same criteria will most likely still be applicable, unless you have created the rule in Garnishments Rules Tables between upgrade passes.

- The Used field on this page may have multiple values.

This field is display only because it keeps track of why data is inserted, whether data passed through conversion, and whether data updated the real garnishment order or not. Review translate values for this field to see details about each value. The rows created by the first section of the report will have the value *M* in the Used field.

- In the following pass, the Used field will have a different value, depending on whether it actually updated the real order or not.

The conversion program will update the value of this field to *U* when the orders are updated, which will happen if the New Law Source and New Rule ID fields have any values. The conversion program will update the value of this field to *X* when the orders are not updated, which will happen if the New Law Source and New Rule ID fields do not have any values.

- The Used field will be very important after running the final upgrade pass.

If there is any row in the data with the value *M* in the Used field, you will have to access the garnishment order and do manual updates where applicable, because there will be no more upgrade conversion after the last validation. These entries would be created by the last validation report and most likely would be for the garnishment orders that were newly created in the last Copy of Production since the last time you took the copy.

## Updating Garnishment Orders After Final Move to Production Pass

To manually update garnishment orders after the final Move To Production pass:

1. Sign on to the Copy of Production database.
2. Select Payroll for North America, Employee Pay Data <country>, Deductions, Create Garnishments where <country> is, for example, CAN, USA, or USF.
3. Enter the employee ID and company that needs manual updates.
4. Click Search.

5. If there is more than one garnishment order for the same employee, select the row with the garnishment ID that needs to be manually updated.
6. Select the Garnishment Spec Data 6 tab.
7. If there is more than one assignment applicable to this garnishment, select the row with the rule that needs to be manually updated.
8. Update the following fields, as needed: Law Source, Rule ID.
9. Click Save.

## Adding Inactive Garnishment Orders

To add inactive orders:

1. Sign on to the Copy of Production database.
2. Select Set Up HRMS, Upgrade, Define Defaults, Upgrade EE Garnishment Orders.
3. Select the Upgrade Garn Rule tab.
4. Click the “+” button to insert a new row.
5. Enter EmplID, Company, GarnishmentID, Law Source, and RuleID from the Inactive Orders section of the report for each order you want to update.

Enter values in the New Law Source and New Rule ID fields from the available prompt lists, selecting the correct value for each order you are updating.

6. Click Save.

There are several things to keep in mind when you are adding new data on this page:

- All manually inserted rows will initially have the value *O* in the Used field, marking them as online entry.
- The system will not prevent you from entering any combination of data on this page.

It does not check whether you are entering only inactive orders. It is your responsibility to decide what updates you are making and add corresponding data on this page.

- In the following pass, the Used field will have a different value, depending on whether it actually updated the real order or not.

The conversion program will update the value of this field to *U* when the orders are updated, which will happen if the New Law Source and New Rule ID fields have any values. The conversion program will update the value of this field to *X* when the orders are not updated, which will happen if the New Law Source or New Rule ID fields do not have any values.

- The Used field will be very important after running the final upgrade pass.

If there is any row in the data with the value *O* in the Used field, you will have to access the garnishment order and do manual updates where applicable, because there will be no more upgrade conversion after the last validation. Follow the above instructions on how to access the garnishment order and fix the data manually.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Payroll for North America	All	All

## Task 5-21: Updating the Employee ID Length

Following data conversion, you will need to review the EMPLID Field Length value in your new release before you use any transaction that creates new employee IDs. This field is used by the process that automatically generates new employee ID values. You should verify that the displayed value is equal to or greater than the length you have used for employee IDs in your previous release.

To review the EMPLID Field Length value:

1. Select Set Up HRMS, Install, Installation Table.
2. Select the Product Specific tab.

The screenshot shows the 'Product Specific' tab in the HRMS installation table. The 'EMPLID Field Length' is set to 4. The 'Save' button is highlighted.

Products | HRMS Options | **Product Specific** | Country Specific | Last ID Assigned | Third Party/System | D

**Benefits Functions**

- ☐ FMLA Administration
- ☒ COBRA Administration
- ☒ Benefits Billing
- ☐ Retroactive Benefits/Deduction
- ☒ FSA Claims Administration

Start Date for BenAdmin: 01/01/1995

**Benefits Deduction Class Order**

- ☒ EE After Tax Then PTax Benefit
- ☐ PTax Benefit Then EE After Tax

**Federal Functionality**

\*Default Pay Basis: Per Annum

☐ NFC Indicator

**NA Payroll / Payroll Interface**

- ☐ Concurrent Calc/Confirm
- ☒ Automatic Employee Tax Data
- ☒ Use State Residence for Local

**T&L / NA Payroll Paysheet Opt**

- ☐ Change Final Check
- ☐ Change Online Check
- ☐ Change Reversal Adjustments
- ☐ Load in Preliminary Calc

**AP Invoice Number Prefix**

AP Inv. Prefix: P

**Commitment Accounting**

- ☒ Encumbrance Trigger

**Human Resources**

- ☒ Multi-Company Organization
- ☐ Automatic Job Termination
- ☐ Automatic Job Suspension

\*Temporary SSN Mask: 999

EMPLID Field Length: 4

**General Ledger**

GL Version: N/A

[Update Version](#)

**Combination Editing Caching**

- ☐ Cache Validations

Cache Retention Days: 0

[Save](#) [Notify](#)

[Products](#) | [HRMS Options](#) | [Product Specific](#) | [Country Specific](#) | [Last ID Assigned](#) | [Third Party/System](#) | [Alternate Character](#)

Product Specific tab

3. Update the EMPLID Field Length value, as needed
4. Click the Save button.

**Note.** You must restart the application server for any changed settings to take effect.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-22: Setting Up Student Group Security

User ID security for student groups is a new feature in PeopleSoft 9.0. It must be enabled in order to access student groups created prior to PeopleSoft 9.0.

For information on setting up student group security, see *PeopleSoft Enterprise Campus Community Fundamentals 9.0 PeopleBook*, “Setting Up Student Groups,” Setting Up a User ID’s Student Group Security.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	Student Administration	All	All

---

## Task 5-23: Reviewing PeopleTools Functionality

The PeopleSoft Enterprise PeopleBooks detail the current PeopleTools functionality. There are many new features delivered in the new release that you may want to use. You should now review the PeopleSoft Enterprise PeopleBooks and PeopleTools Installation Guide to configure your environment properly. This may include, but is not limited to, configuring and starting a process scheduler, report server, and reviewing portal settings.

See the Enterprise PeopleTools installation guide for your database platform on your new release.

See PeopleSoft Customer Connection (Implement Optimize + Upgrade, Upgrade Guide, Upgrade Documentation and Software, Release Notes).

You should review the following considerations:

- If you applied a PeopleTools patch earlier in the upgrade, review the patch documentation and run any steps that you have not already performed during the upgrade.

Check your Change Assistant output directory if you do not know whether a script was already run during the upgrade process.

- You must reimplement any workflow that you want to use in the new release.
- PeopleTools has updated the styles that define the look of the user interface.

The classic (old) style sheet, as well as two new styles, are delivered with this release of PeopleTools. PeopleTools system databases and PeopleSoft 8.4 applications use the classic style, whereas all other

applications use the new dark blue style. After the PeopleTools portion of the upgrade, PeopleSoft sets the classic style as the default, but you can update to one of the new user interface styles.

See Appendix: “Changing the User Interface.”

---

**Note.** The new user interface styles are supported by Internet Explorer release 5 and above and Netscape Navigator release 6 and above. If you are using any other browser or release, the system uses the classic style as the default.

---

- PeopleTools uses Verity release 5.0 to implement free text search.

Verity 5.0 is not compatible with the version of Verity that was used in previous PeopleTools releases. Check the necessary application patches that may be required in order to use the new version of Verity.

See PeopleSoft Customer Connection (Updates + Fixes, Required for Install or Upgrade).

- An upgrade may cause the web server domain name, port number, or servlet path required to access Web Server resources to change.

As a result, performing PS/nVision drill-down operations on reports that were created prior to upgrade would fail. This is mainly because drill-down hyperlinks are by design hard coded into PS/nVision reports. A simple search and replace utility has been provided that can be used to replace old hyperlinks with new ones. This Microsoft Excel macro is located at %PS\_HOME%\EXCEL\UpdateNvsDrill.xls.

- Integration Broker was rewritten in PeopleTools 8.48.

If you use Integration Broker, you will need to perform setup configuration and review the explanation of metadata mapping.

See Enterprise PeopleTools 8.48 PeopleBook: PeopleSoft Integration Broker.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 5-24: Preparing the Content Provider Registry

You should perform this task if you use Enterprise Portal 8.4 or higher running on PeopleTools 8.4x with full or partial navigation load access method. This means you do not use a single link to access your content provider databases, but instead you load some or all of the portal registry structures from the content provider database into your Enterprise Portal database. PeopleSoft refers to Content Provider databases as the application databases that contain the transaction content. Your Copy of Production database is your Content Provider database for this task.

When you upgrade a content provider database, the registry structures are updated, old registry structures are removed, and new registry structures are added. These changes need to be copied to the Enterprise Portal database by updating the portal registry structures in your Enterprise Portal database to match what is in the Content Provider database. Follow the detailed instructions in the appendix referenced below.

See Appendix: “Upgrading the Content Provider Registry.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 5-25: Updating the Portal Options Data

In this step you update the PeopleTools Portal Options data.

**Note.** Only perform this step if your upgraded database is on PeopleTools 8.46 or greater.

This step sets the portal options prefix and Owner ID. These values are used when creating Pagelet Wizard definitions and Navigation Collection objects.

To set the Portal Options Prefix and Owner ID:

1. From your browser, sign on to your New Copy of Production database.
2. Select PeopleTools, Portal, Portal Utilities, System Options.
3. Update the value for the Registry Object Prefix with a 1- to 4-character prefix that is unique to your organization.

**Note.** Do *not* use PAPP, PAPX, PAPQ, PAPI, PRTL, EO, or PT. Do *not* use any product line specific prefix (such as CR, HC, EP, or CI). Do *not* use a blank value.

4. Enter the Owner ID value with your organization's specific owner ID.

**Note.** The Owner ID is a translate value on the PeopleTools field OBJECTOWNERID. Do *not* use any delivered product Owner ID. If you do not have an Owner ID, then either create one, or leave the Owner ID value as a blank space.

5. Click Save.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

## Task 5-26: Stamping the Database

In this step, you set the database to the release level of the Demo database. The values that you enter here appear whenever you view the Help, About PeopleTools dialog.

To stamp the database:

1. Launch Application Designer on your Copy of Production database using the new PeopleSoft release.
2. Select Tools, Upgrade, Stamp Database.
3. Fill in all three of the PeopleSoft Release fields with the appropriate value for your product line and release number:

HRMS and Campus Solutions, 9.00

4. In the Service Pack field, enter the service pack number to which you are upgrading. For example, if you are upgrading to SP2, enter the number 2. If you are upgrading to a release that is not at a service pack level, enter 0.
5. Click Stamp.
6. Close Application Designer.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

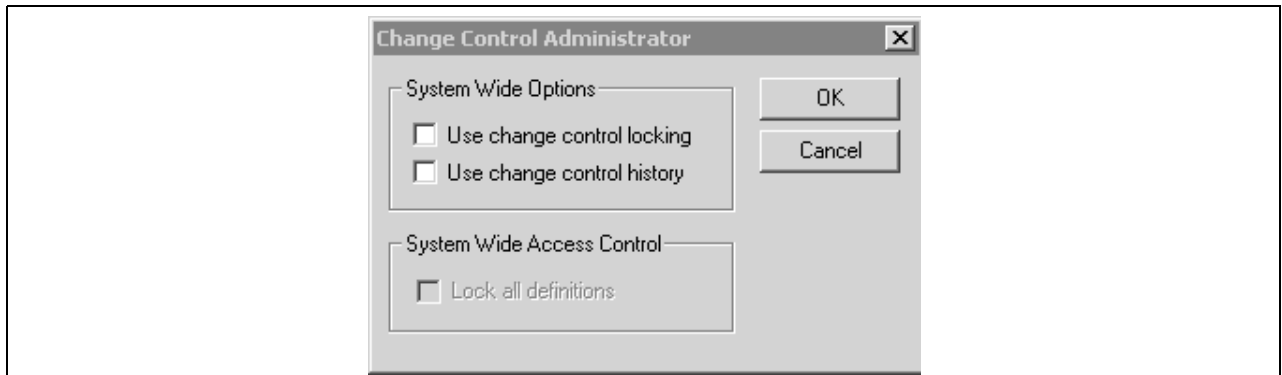
## Task 5-27: Reviewing Change Control

Earlier in the upgrade process, in the beginning of the chapter “Apply PeopleTools Changes,” the Change Control feature was disabled. In this step, you re-enable Change Control, if your site uses this functionality.

To turn on Change Control:

1. Sign on to the Target database using Application Designer.
2. Select Tools, Change Control, Administrator.

The following dialog box appears:



Change Control Administrator dialog box

3. Set Use change control locking and Use change control history according to your site specifications.



---

**Note.** Move to Production: The Change Control feature slows down copy functions. The large copy projects are only executed during the initial pass, and the feature is only disabled during the initial pass. If you enable the feature at this point, it will remain enabled during future test Move to Production passes.

---

See “Apply PeopleTools Changes,” Turning Off Change Control.

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Initial	All	All	All

---

## Task 5-28: Backing Up Before Testing

Back up your Copy of Production database now. This enables you to restart your upgrade from this point, should you experience any database integrity problems during the remaining tasks in the upgrade process.

See Appendix: “Planning for Upgrade Testing.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 5-29: Testing Your Copy of Production

In this task, you test your Copy of Production. Testing your Copy of Production will ensure that you can still operate your day-to-day processes on your new release. After you have reviewed your DDDAUDIT and SYSAUDIT, verify that the system is working properly by reviewing the system online. After you are comfortable that the system is working properly, you can perform the Test Move to Production upgrade pass.

See Appendix: “Planning for Upgrade Testing.”

### Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All



## CHAPTER 6

# Apply Changes to Production Database

This chapter discusses:

- Understanding the Move to Production
- Testing the Move to Production
- Testing Once More
- Performing the Move to Production
- Completing the Upgrade Survey

---

## Understanding the Move to Production

Once you complete all of the necessary tasks to launch your system into production, you are ready to begin your Test Move to Production passes or to move your system into production.

---

## Task 6-1: Testing the Move to Production

This section discusses:

- Understanding the Move to Production Test
- Understand Move to Production
- Creating a New Change Assistant Job

## Understanding the Move to Production Test

Everything you have done to this point is the initial pass of the upgrade process. Now you are ready to start the Test Move to Production pass. The initial pass is very time consuming and requires a lot of analysis at different steps of the process and to troubleshoot issues. The Test Move to Production pass is a different series of steps, which includes a subset of the previous tasks, and takes advantage of the tasks performed during the first upgrade pass.

You should perform as many test moves to production as necessary to work out any issues and to be comfortable with the process. During each Test Move to Production you will be able to refine the process so that you can save time and avoid manual processes. These test passes will also let you know how long the process takes so you can plan your production downtime for your move to production weekend.

## Task 6-1-1: Understand Move to Production

The following text is a high level view of what you will be doing in the Move to Production test pass. The remaining steps in this task will prepare your test environment. For example, you may need to move some scripts generated in the initial pass to a new Change Assistant staging directory. Next you will create a new Change Assistant job, setting the Type of Upgrade to Move to Production. That will give you a job with steps filtered with only those steps that apply to the Move to Production (MTP) test pass. From that point forward, you will simply follow the steps as they exist in your new job.

One of those first steps will be to take a Copy of Production. This second Copy of Production is sometimes referred to as the “New Copy of Production.” The first Copy of Production, or “old” Copy of Production, will now be the Source database (it was the Target database in the initial test pass). The New Copy of Production is now the Target database.

The steps executed in the MTP pass vary in several ways. Many of the tasks and steps in the initial test pass will be replaced in the MTP pass with Data Mover export and import scripts. In the initial pass, some steps required you to make functional decisions and take time to manually set up data. That data can be copied from the first database to the next, saving you setup time and eliminating the chance for manual error or typos.

Also, the MTP pass does not repeat the database compare/copy steps. You made the decisions once; there is no need to repeat these steps. Instead, a Data Mover script, MVPRDEXP, will export all the tables that contain the PeopleTools objects like records and PeopleCode from the first database. Another Data Mover script, MVPRDIMP, will import those tables into the second database. Anything you have done to PeopleTools objects while executing or testing the first pass—copied objects from DMO, reapplied customizations, applied updates from the PeopleSoft Customer Connection website—will be moved to the second Copy of Production with these scripts.

Another important difference with the MTP pass is the handling of SQL scripts that create and alter tables. In the initial pass, you generated the SQL scripts, sometimes edited the SQL script, and then executed the SQL scripts. In the MTP pass, you may be able to skip the generation steps and use the SQL you previously generated. This is another way to save time in your critical go-live window and is the ultimate goal, but it is an incremental process to get to that point.

In the first MTP pass, everyone must regenerate the SQL. There are small differences between the initial and MTP passes that require the SQL to be regenerated in at least one MTP pass. The Change Assistant Templates are delivered with the steps set this way.

In subsequent MTP passes, you may choose to “turn off” the generation steps if possible. If you have not changed any records at the end of one MTP pass, then you can reuse the SQL in your next pass. If you have done anything to change records, you should generate SQL again. This can include things like applying PeopleTools upgrades (for example, 8.47 to 8.48), or applying updates from the PeopleSoft Customer Connection website that involve record changes, or making additional customizations to records.

If you choose to skip some of these steps, do one of the following: mark the step complete in your job, or change the step properties in the template, so the step will never show up in your MTP filtered job again. To change the step properties, double-click on the step to open the Step Properties dialog, and change the Type of Upgrade to Initial Upgrade. In addition, copy the SQL scripts from the previous pass output directory to the new pass output directory. Change Assistant will look for the SQL scripts in the output directory set on the job’s Database Configuration, so make sure it will find them when it tries to run them.

The steps that are eligible for this treatment will contain Move To Production documentation notes indicating such.

---

**Note.** If you have made any changes to your trees, tree structures, or PS/Query objects since the upgrade began, you may want information on how to preserve those changes.

---

See Appendix: “Preserving Queries and Tree Objects.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

## Task 6-1-2: Creating a New Change Assistant Job

You need to create a new Change Assistant Job for each test Move to Production pass.

To create a new Change Assistant job:

1. Oracle recommends that you use new output and stage directories for each new test pass. Create those directories now.
2. From Change Assistant, select Tools, Options and specify the new output and staging directories on the Change Assistant Options page.
3. Select File, Open Environment and select the environment.
4. Review the configuration on the General Settings window.

The Database Type, Language and SQL Query Executable will be the same as your previous job. Make changes to the PS\_HOME settings, if necessary, and select Next.

5. Specify the Source Database setup information and click Next.

This is the Copy of Production database from your previous pass.

6. Specify the Target Database setup information and click Next.

This is the new Copy of Production database.

7. Review the environment configuration on the Confirm Selections window and click Next to save the changes to the environment.
8. Select File, New Job.
9. Select the template on the Use Template window and click OK.
10. On the Type of Upgrade window, select Move to Production.
11. Click OK.

A new upgrade job using the naming convention of *Template\_Environment\_Move to Production* is created.

12. Highlight the job name and select Edit, Set Documentation Directory, then select the directory where the documentation is located and click OK.

If you have been using documentation printed from the filtered view on the jobs, print the documentation again. The steps for Move to Production are different than in the initial pass.

13. Select View, Documentation.
14. Select View, Expand All to display all the steps in the job that apply to your upgrade.

The job will contain steps that were not in the initial upgrade pass and will exclude some steps that were in the initial upgrade pass based on the step properties.

Now you are ready to run the job.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 6-2: Testing Once More

As in any implementation project, you must consider planning, resources, development, and training. Testing also needs to be an integral part of your implementation project. Testing your database once more, after you have completed the upgrade, ensures that you can still operate your day-to-day processes on your new PeopleSoft release.

The level of testing in this task will focus primarily on the strategies to employ before moving into production.

See Appendix: “Planning for Upgrade Testing.”

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	Both	All	All	All

---

## Task 6-3: Performing the Move to Production

When you are ready, you can move the system into production. Take your system out of production and perform all of the steps involved in testing the Move to Production against your production database.

See Testing the Move to Production.

## Properties

Database Orientation	Initial or MTP	Products	Platforms	Languages
Target	MTP	All	All	All

---

## Task 6-4: Completing the Upgrade Survey

We are interested in feedback on your upgrade experience and any thoughts and/or suggestions you have on how we can improve the process in the future. Note that this survey should only be accessed once you have completed your upgrade and are in production on the new release.

**See Also**

<http://www.peoplesoft.com/go/upgradesurvey>

**Properties**

<b>Database Orientation</b>	<b>Initial or MTP</b>	<b>Products</b>	<b>Platforms</b>	<b>Languages</b>
Target	MTP	All	All	All





## CHAPTER 7

# Appendices

---

### Understanding Appendices

The appendices portion of this documentation contains information you may need for your upgrade. The appendices have been referenced throughout the upgrade documentation for further understanding of the upgrade you are performing. Oracle recommends that you read each appendix as it is referenced in the documentation.



## APPENDIX A

# Applying Fixes Required for Upgrade

This appendix discusses:

- Preparing to Apply Fixes
- Applying Fixes During Installation
- Applying Fixes After Copying Project
- Applying Fixes After Data Conversion
- Applying Fixes Between Upgrade Passes
- Applying Fixes in Move to Production

---

### Task A-1: Preparing to Apply Fixes

This appendix gives general instructions for applying a Required for Upgrade fix for your upgrade. If the directions given in a particular fix are different from those given here, then follow the instructions in the fix.

It is important that you run your upgrade using the latest versions of all upgrade software. In Customer Connection, check the upgrade page and the Updates and Fixes page to ensure that you have all of the latest code.

---

**Note.** Because of interdependencies between products and product lines, you will need to download all fixes for all HRMS products and product lines, even if you are not licensed for them. If you do not need the program, the upgrade will detect it but you can introduce an error by not applying it. This includes HRMS Commercial, HRMS Education and Government, and HRMS Federal Government product lines, as well as all products within the product lines.

---

Ideally, you should follow the steps below to apply the various files and fixes.

To apply files and fixes:

1. Install the new release from the CD.
2. Apply any additional scripts and projects from the Customer Connection upgrade page to your new release codeline (and to the New Release Demo database, if applicable).
3. Apply any other Required for Upgrade fixes from Customer Connection's Updates and Fixes to your new release codeline (and to the New Release Demo database, if applicable).
4. Run your initial pass of the upgrade.
5. Before you begin each subsequent upgrade pass, check the upgrade page for new versions of any files you previously applied.

Then check Updates and Fixes for any new Required for Upgrade fixes.

Your initial upgrade pass will differ from your subsequent Test Move to Production passes. Some of the upgrade tasks and steps are common to both the initial upgrade pass and the Move to Production pass. For this reason, you may find Required for Upgrade fixes that do not apply to the upgrade pass that you are currently performing. The details provided with each fix will help you determine whether to apply the fix and when to apply it. The fix will also tell you what to do if you have already passed the step for which the fix is needed.

How you apply a fix depends on where you are in the upgrade process. This appendix explains how to apply a typical fix, and is organized by the various points within the upgrade where you will apply fixes.

---

## Task A-2: Applying Fixes During Installation

In the chapter, “Install the Software,” of *Getting Started on Your PeopleSoft Upgrade*, you should first download and apply all files and objects from the upgrade page on PeopleSoft Customer Connection. Then you must download all Required for Upgrade fixes from Updates and Fixes on PeopleSoft Customer Connection. You can use the instructions in this section to apply any additional fixes that are posted, until you reach the task, “Running New Release Compare Reports.”

If a fix contains a project that needs to be copied from a file, apply it to your New Release Demo database during installation. If the project contains changes for records or fields, those objects will be updated during the normal compare and copy steps in the upgrade. You will not have to build objects in the project separately or consider whether it will have an impact on customizations. You will do that with the rest of the objects during the upgrade. Apply as many of the fixes as you can at this time.

To apply script fixes during installation:

1. Download Required for Upgrade change packages using the “Download Change Package” functionality in Change Assistant.
2. Use Change Assistant to apply the updates into your New Release Demo database.

Review the documentation included with each update prior to applying each update. There may be manual steps that need to be performed in order to successfully apply the update.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your current release, “Applying Updates.”

---

## Task A-3: Applying Fixes After Copying Project

It is best not to apply fixes during the compare and copy tasks in the “Run and Review Compare Reports” and “Apply Application Changes” chapters of the initial upgrade pass. It can also be cumbersome to apply record and field changes during the creating and altering of tables in the “Complete Database Changes” chapter. It is, therefore, best to wait until just before the “Running Data Conversions” task in the “Apply Application Changes” chapter to apply additional fixes. Most of the fixed objects will be data conversion code, delivered in projects.

To apply PeopleSoft project fixes before data conversion:

1. Download Required for Upgrade change packages using the “Download Change Package” functionality in Change Assistant.
2. Use Change Assistant to apply the updates into your New Release Demo database for this upgrade pass.

Review the documentation included with each update prior to applying each update.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your current release, “Applying Updates.”

3. The project is now loaded on your New Release Demo database. You should run a project compare to make sure that the objects in the fix will not overwrite any of your customizations.

If you find customizations, you must decide how to deal with them before you copy the fix to your Copy of Production.

4. If you are performing a Move to Production upgrade pass, first migrate the change packages into the Source database for this upgrade pass.

If needed, first set up Change Assistant with the environment information for your Source database. If you customized any of the objects delivered in the change package, you should repackage the fix to include your customizations. If you did not customize any objects delivered in the fix you may directly apply them to your Source database.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your current release, “Applying Updates.”

5. Migrate the change packages into the Target database for this upgrade pass.

If needed, first set up Change Assistant with the environment information for your Target database.

---

## Task A-4: Applying Fixes After Data Conversion

At this point, you have already converted all of your data for the upgrade pass, and you cannot apply Application Engine program fixes and use them in this upgrade pass. You should refer to the fix instructions to determine what to do in each case. Often, the instructions say that you need to restore your database from a pre-conversion backup and rerun data conversion to get the benefits of the fix. Because this is the only way you can get the fix onto your current Copy of Production, you may decide to allow the error and not apply the fix until you do a Test Move to Production. Then after you have completed that test pass, you can test the affected function. However, you should not do this if your next pass is your final Move to Production, and you are going into production with the resulting database. You should always test your upgraded database between test passes if changes have been made to procedures, scripts, or programs. You do not want any surprises during the final Move to Production.

---

## Task A-5: Applying Fixes Between Upgrade Passes

You can apply fixes just before you start a Test Move to Production pass in the same way you would in the step above, Apply Fixes After Copying Project. In those instructions, you apply the fix to your New Release Demo database and compare it to the Copy of Production. Make sure that you do the database comparison in order to verify that the fix does not wipe out any customizations you made to Application Engine programs during your initial upgrade pass. If you have made customizations, merge your customizations into the new Application Engine code on the New Release Demo database. Then apply the fix to your Copy of Production, which you will use as the Source database in the Test Move to Production. The fix will then get moved to your New Copy of Production when you run the MVPRDEXP.DMS and MVPRDIMP.DMS scripts in the “Apply PeopleTools Changes” chapter.

---

## Task A-6: Applying Fixes in Move to Production

Once you have started a Test Move to Production, do not apply any fixes until just before data conversion. Apply any fixes using the previous step, “Applying Fixes After Copying Project.” In those instructions you apply the fix to your New Release Demo database and compare it to your Copy of Production. Instead of using the original Copy of Production as the Target, you must now use your New Copy of Production, the one defined as the Target in your Move to Production Change Assistant job. Be sure to do the database comparison to verify that the fix does not wipe out any customizations that you made to Application Engine programs during your initial upgrade pass. If you have made customizations, merge your customizations into the new Application Engine code on the New Release Demo database, then copy the project to your New Copy of Production.

## APPENDIX B

# Changing the User Interface

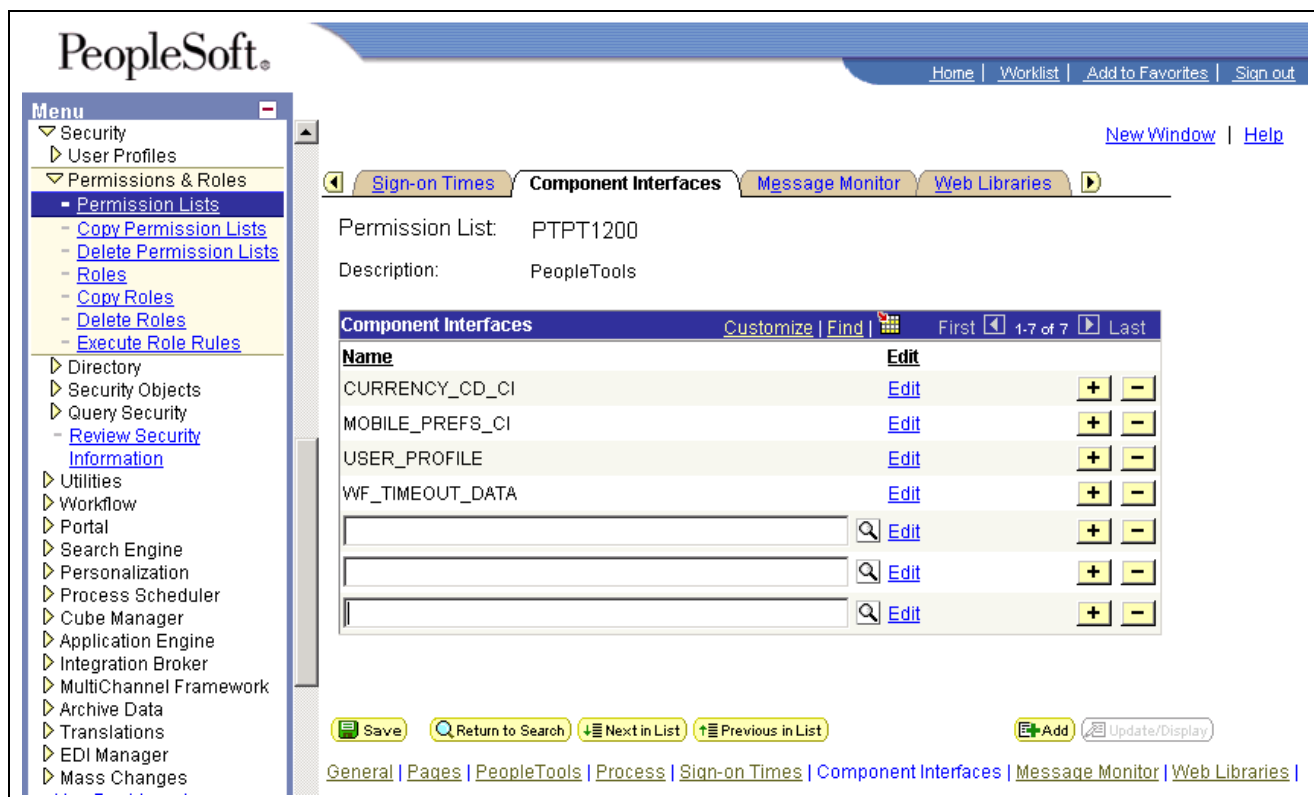
This appendix discusses:

- Change the User Interface
- Change the User Interface for Mobile

## Task B-1: Change the User Interface

Three user interface options were delivered with your current release of PeopleSoft 8.x. The classic style, the style seen in previous releases of your PeopleSoft application, is set as your default style. The following are examples of the three delivered styles: classic, light blue, and dark blue.

The classic style, shown below, is set as your default style delivered with your PeopleSoft release.



The classic style user interface option

The following example represents the light blue style.

The screenshot shows the PeopleSoft user interface in a light blue style. The top navigation bar includes links for Home, Worklist, Add to Favorites, and Sign out. The left-hand menu is expanded, showing the 'Permissions & Roles' section with sub-items like 'Permission Lists', 'Copy Permission Lists', 'Delete Permission Lists', 'Roles', 'Copy Roles', 'Delete Roles', and 'Execute Role Rules'. The main content area displays the 'Component Interfaces' tab for the 'PTPT1200' permission list, which is described as 'PeopleTools'. A table lists several component interfaces: CURRENCY\_CD\_CI, MOBILE\_PREFS\_CI, USER\_PROFILE, and WF\_TIMEOUT\_DATA, each with an 'Edit' link and '+' and '-' buttons. At the bottom, there are buttons for Save, Return to Search, Next in List, Previous in List, Add, and Update/Display. The breadcrumb trail at the very bottom reads: General | Pages | PeopleTools | Process | Sign-on Times | Component Interfaces | Message Monitor | Web Libraries |

The light blue style user interface option

This example represents the dark blue style.

This screenshot shows the same PeopleSoft user interface but in a dark blue style. The layout and content are identical to the light blue version, including the top navigation bar, the left-hand menu, the 'Component Interfaces' tab for 'PTPT1200', the table of interfaces, and the bottom buttons and breadcrumb trail. The primary color scheme is a darker blue, providing a different visual theme for the application.

The dark blue style user interface option



See the Enterprise PeopleTools PeopleBook: PeopleSoft Application Designer for your new release.

To change your styles, you must delete the substyle sheets associated with the classic style and replace them with either the light or dark blue substyle sheet.

---

**Note.** The new user interface looks are supported by Internet Explorer release 5 and above and Netscape Navigator release 6 and above. If using a browser and release other than these, the system defaults to the classic style.

---

To enable one of the new user interface looks:

1. In Application Designer, open the stylesheet PSSTYLEDEF.
2. Select the PSALTERNATE Sub Style Sheet, and click Delete.
3. Select Insert, Insert Sub Style Sheet.
4. Select PSALTERNATE\_LIGHTBLUE or PSALTERNATE\_DARKBLUE.
5. Repeat steps 1 through 4 for the PTSTYLEDEF and PSACE Sub Style Sheets, making sure to use the same extension (\_LIGHTBLUE or \_DARKBLUE) as you did for PSALTERNATE.
6. Select Save.
7. Open the stylesheet PSQUERYSTYLEDEF.
8. Select the PTQUERYSTYLESUB Sub Style Sheet, and click Delete.
9. Select Insert, Insert Sub Style Sheet.
10. Select PTQUERYSTYLESUB\_LIGHTBLUE or PTQUERYSTYLESUB\_DARKBLUE.  
Use the same extension as you did in Step 4.
11. Click Save.

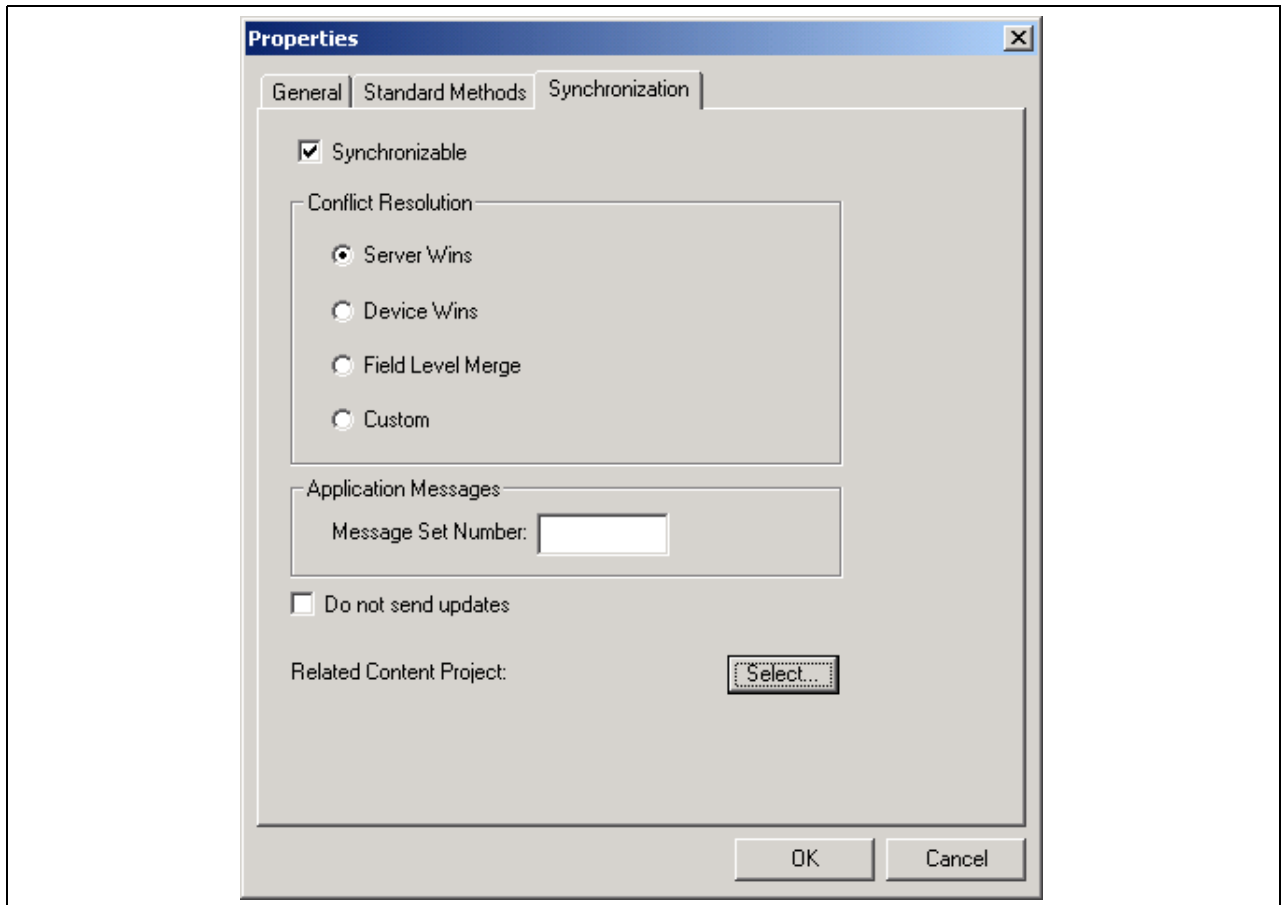
---

## Task B-2: Change the User Interface for Mobile

You use related content projects to synchronize extraneous PeopleTools objects to a mobile device. In particular, you can use a related content project to synchronize a new or customized stylesheet to the mobile device. Related content projects are associated with a component interface. PeopleSoft recommends that you associate a given related content project with a synchronizable component interface that is common to your mobile application.

To change the user interface look for mobile applications:

1. Follow the procedure in the previous task, “Change the User Interface,” to add the new or customized stylesheet to a PeopleTools project.
2. In Application Designer, open a synchronizable component interface that is common to your mobile application.
3. Open the Properties dialog box from the pop-up menu.
4. Select the Synchronization tab.



The Synchronization tab

5. Click Select at the Related Content Project field.
6. Find and highlight the desired project and click Select.
7. Click OK.

The stylesheet will be synchronized to the mobile device during the next bootstrap or update applications synchronization.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Component Interfaces for your new release.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Mobile Agent for your new release.

## APPENDIX C

# Improving Performance

---

## Understanding Performance Improvement

Performance is always a challenge when you are upgrading. Much of the data currently in your database will be affected. No other batch processing works quite like it. Upgrade performance is sensitive to your unique environment and data. These performance recommendations are designed to help you improve performance during your upgrade.

---

## Task C-1: Planning Your Upgrade for Performance

Review the following guidelines to help plan for better upgrade performance:

- Provide as much hardware, memory, and disk space as you can.
- Run long processes on a dedicated server, not the client. Configure that server similarly to your production environment.
- Size the Copy of Production database (Target) like your production database — allow for growth.
- Use the appendix titled “Sizing Tables for the Upgrade” to determine how much space you will need. These recommendations are minimum levels; you should plan to use more.
- Use a dedicated workstation, configured like the production environment.
- Provide an application server and process scheduler for the Target database on the new PeopleSoft release.
- Look for fragmented tables. Resize or reorganize the initial and next extends accordingly.
- Know which tables are your largest. This information will be valuable during the upgrade.

---

## Task C-2: Performing the Upgrade

This section discusses:

- Verifying the Database Structure Task
- Creating and Altering SQL Scripts
- Performing Data Conversion

## Task C-2-1: Verifying the Database Structure Task

When performing the “Modifying the Database Structure” task, perform all recommended update statistics, so the optimizer can make a good decision.

## Task C-2-2: Creating and Altering SQL Scripts

The following list provides tips to use when performing the “Modifying the Database Structure” task.

- Perform all recommended update statistics, so the optimizer can make a good decision.
- Look for large tables that are being altered. Create separate scripts for each of these tables and run the script concurrently with your other alter scripts. You can run these concurrently even if you are using the Alter by Rename build option because each table creation uses an independent temporary table name in the new PeopleSoft release (*PSYrecname*).
- Run your create table script concurrently with the alter tables without deletes script.
- Some indexes will not create during alter without deletes because of a unique key violation. You can expect this because index structures have changed. The data in these indexes will be corrected during data conversion. You can create the index as a non-unique key for data conversion. Create this non-unique index, without the field that caused the error, so that data conversion does not have to update the index as well as the data pages as it corrects the data in this field.

## Task C-2-3: Performing Data Conversion

Oracle delivers the Change Assistant templates to run the steps on the client by default. The server can run Application Engine programs. You may want to run these programs on the server to increase performance. Change Assistant uses the PSEMHUB and PSEMAgent to execute jobs to the server. You will need to configure and start your Environment Management Hub and Agent, enter the Environment Management information into the Change Assistant options, set your Change Assistant mode, select the Perform Server Processing check box, and modify the run location of the steps you wish to run on the server. Consider running data conversion application engine programs concurrently as they were designed with no dependencies between programs. Any exceptions to this are noted in the documentation. If you would like to take advantage of concurrent processing, modify the step properties and set the Run Concurrently option to Yes.

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release.

---

## Task C-3: Improving Performance for Your Platform

This section discusses:

- Using Oracle
- Using DB2

### Task C-3-1: Using Oracle

You should use the cost-based optimizer during the upgrade. The application engine scripts use MetaSQL to run the update statistics command at various times during data conversion. If you are running the rule-based optimizer, you will not take advantage of these statements.

When running the alter scripts, remember the Oracle hint `/*APPEND*/`. This command will improve the performance of your alter script if you are doing an “Alter by Rename.” This hint performs like an Oracle Direct Load Insert and does not capture redo or recovery information. The syntax is as follows:

```
INSERT /*+ APPEND */ INTO TABLENAME (FIELD1, FIELD2, ...
```

Make sure your database administrator has turned autoextend on. Autoextend allows tablespaces to grow larger than their set maximum size and will be useful during the upgrade process since tablespaces grow several times larger than they would in production.

Alter the tablespace for PSIMAGE and increase it to 200 MB; autoextend on the next 10 MB; set the maxsize to *unlimited*.

## Task C-3-2: Using DB2

Perform all recommended update statistics on the system catalog as well as the application tablespaces.

Interspersed throughout the Application Engine data conversion programs are steps that make a dynamic call for RUNSTATS to be performed against selected tables. An example of how this is used follows:

- Step01 - Start with an empty Temp\_Table\_TAO.
- Step02 - INSERT 60,000 rows into the Temp\_Table\_TAO.
- Step03 - Application Engine Step calls %UpdateStats(Temp\_Table\_TAO).
- Step04 - You can now perform SQL against Temp\_Table\_TAO using new statistics.

This will allow you to use indexes when processing against Temp\_Table\_TAO. If DB2 z/OS is not configured correctly, Application Engine programs skip these steps and the performance of the SQL steps that follow will suffer.

In order to take advantage of the %UpdateStats feature, at least one of these conditions must be met:

- You implemented the DSNUTILS stored procedure.
- You followed the enhanced installation path.

See the Enterprise PeopleTools Installation for DB2 UDB for z/OS for your new release.

---

**Note.** If you disable the %UpdateStats feature of Application Engine, it will adversely affect the performance of your data conversion programs, because statistics will not update when the tables are loaded with large volumes of data.

---

## Task C-4: Consulting the Global Support Center

If you do have a problem with your upgrade, contact the Global Support Center (GSC). Oracle will be able to give you a solution to the problem faster if you supply the following information:

- Include details about the table row counts and indexes available on the tables involved in the processing:
  - Include indexes in your physical database, not those defined in Application Designer.
  - Mention any additional indexes that you custom-added; they could be getting in the way.
- Include RDBMS (Oracle, SQL Server, or DB2, and so on) and RDBMS release (for example, Oracle 7.3 or 8.16).
- If you are running on Oracle, specify whether you are you running in cost-based or rule-based mode.

- Include the PeopleSoft upgrade path for both PeopleTools and the application.
- Provide trace files: PeopleTools trace and RDBMS-specific trace files, SQL explains, and so on.

## APPENDIX D

# Modifying the Approval Rule Set Program

This appendix discusses:

- Modify Approval Rule Set Related Tables

---

### Task D-1: Modify Approval Rule Set Related Tables

This Application Engine program is used once during the upgrade process. The program alters some approval rule set-related tables.

To modify approval rule set-related tables:

1. Make sure the user running this program has security rights to create tables on the database.
2. Verify and modify, if necessary, the storage areas used in the Application Engine steps indicated below for tables and indexes.

DB2 z/OS platform users :

- Review the values for PRIQTY and SECQTY and modify them if necessary.
- Review the DBNAME and modify it if necessary.

The DDL selects the value for DBNAME from the following SQL:

```
SELECT PARMVALUE FROM PSDDLDEFPARMS WHERE PLATFORMID = 1 AND STATEMENT_TYPE =>
1 AND PARMNAME = 'OWNER'
```

The DDL selects the PARAMVALUE from:

```
SELECT parmvalue FROM psddldefparms WHERE platformid = 1 AND statement_type =>
2 AND parmname = 'STOGROUP'
```

If the value for DBNAME is not the same as the value for OWNERID, you will also need to replace %Bind(DBNAME ,NOQUOTES) with your database name for the Application Engine steps listed in the tables below.

3. These values are used in the %Bind specified in the tables below. Modify this SQL if this is not the case for your database.

Tables Created		Indexes Created	
PSYAPPR_RULE_AMT PSYAPPR_VA2_WRK PSYPSVAITEM		PS_APPR_RULE_AMT PS_PSVAITEM	
Platform	SQL Section DDL statements	Table Creation Storage Area	Index Creation Storage Area
DB2 UNIX/NT	ANSI.Step01 UNICODE.Step01	IN PTTBL INDEX IN PTTBLIDX NOT LOGGED INITIALLY  IN PTWORK INDEX IN PTWORKIDX NOT LOGGED INITIALLY  IN PSIMAGE INDEX IN PSIMAGEIDX NOT LOGGED INITIALLY	No name specified
DB2 z/OS	ANSI.Step01 UNICODE.Step01	%Bind(DBNAME ,NOQUOTES).PTTBL	USING STOGROUP %Bind(PARMVALUE ,NOQUOTES) PRIQTY 720 SECQTY 720 CLUSTER BUFFERPOOL BP3 CLOSE NO
Informix	ANSI.Step01 UNICODE.Step01	PTTBL	PSINDEX



Platform	SQL Section DDL statements	Table Creation Storage Area	Index Creation Storage Area
Oracle	ANSI.Step01 to ANSI.Step14 UNICODE.Step01 to UNICODE.Step14	<p>TABLESPACE PTTBL STORAGE (INITIAL 40000 NEXT 100000 MAXEXTENTS UNLIMITED PCTINCREASE 0) PCTFREE 10 PCTUSED 80</p> <p>TABLESPACE PTWORK STORAGE (INITIAL 40000 NEXT 100000 MAXEXTENTS UNLIMITED PCTINCREASE 0) PCTFREE 10 PCTUSED 80</p> <p>TABLESPACE PSIMAGE STORAGE (INITIAL 40000 NEXT 100000 MAXEXTENTS UNLIMITED PCTINCREASE 0) PCTFREE 10 PCTUSED 80</p>	TABLESPACE PSINDEX STORAGE (INITIAL 10000 NEXT 100000 MAXEXTENTS UNLIMITED PCTINCREASE 0) PCTFREE 10
Sybase SQL Server	ANSI.Step01 UNICODE.Step01	No name specified	No name specified



## APPENDIX E

# Planning for Upgrade Testing

This appendix discusses:

- Understanding Testing Techniques
- Deciding When to Test
- Evaluating Your Testing Requirements
- Defining Your Testing Strategy
- Determining the Testing Conditions
- Developing Your Test Plan
- Developing Test Scripts
- Reviewing Tips and Techniques

---

## Understanding Testing Techniques

As with any project, testing is a critical part of your upgrade project. With proper testing, you can ensure that you upgrade successfully and you are ready for your Move to Production.

Upgrades vary in complexity and scale from release to release and customer to customer, so the testing periods and the activities required to perform testing vary from upgrade to upgrade. Because PeopleSoft cannot anticipate how every organization uses the system to fit their own business practices, including customizations and data setup, PeopleSoft does not deliver upgrade test scripts. However, there are some general testing guidelines that you can follow to assist with your upgrade testing. In this section, you will find information that will help you plan your testing efforts.

---

## Task E-1: Deciding When to Test

An effective testing strategy involves an understanding of the stages of a PeopleSoft upgrade and where, within these stages, testing should be performed. You can take more than one approach and use more than one method to test your upgrade.

---

## Task E-2: Evaluating Your Testing Requirements

To evaluate your testing requirements, you need the following information:

- The number of products and modules you currently have in your production database.
- The number of customizations you have in your production database.
- The functional design and business requirements addressed by each customization.
- Your online, batch, and reporting business processes that you want to include in testing.

---

## Task E-3: Defining Your Testing Strategy

This section discusses:

- Understanding Your Testing Strategy
- Evaluating Unit Testing
- Evaluating System Testing
- Evaluating Integration Testing
- Evaluating Parallel Testing
- Evaluating Performance Testing
- Evaluating User Acceptance Testing
- Evaluating Regression Testing

### Understanding Your Testing Strategy

Once you evaluate your testing requirements, you can determine what types of testing you need. You should define the tests to be performed for the project and the goals of each test—including roles and responsibilities, test-case management, control points, and success criteria. In addition, you should define and document the scope of each type of testing. Use the definitions below to determine the levels of testing required in your organization.

To ensure upgrade success, be sure to train upgrade members before the upgrade. It is critical to have educated testers to ensure adequate test coverage of new functionality.

The testing types below do not necessarily run systematically, one after another. Different test conditions can sometimes run in parallel.

---

**Important!** It is important to test not only your customizations and changes in your new release but also your standard business practices. By doing this you ensure that your normal business practices have not been altered by the new release and/or your customizations.

---

## Task E-3-1: Evaluating Unit Testing

In this stage of testing, you have completed your upgrade tasks and your database is now at the new release level. However, you should unit test before you use the new system. Unit testing validates data, business rules, and business process requirements. In addition, it ensures that business processes work as designed and your database is ready for full functionality testing. The processes for performing unit testing are described below:

- Test individual online transactions and batch processes on the upgraded database.
- Validate data converted during the upgrade.
- Verify that you can access existing data and enter new data successfully.
- Test customizations reapplied to the upgraded database.
- Each customization is tested individually along with all related processes.
- Business processes are not tested.
- Test scripts are not required.
- Test – Document – Resolve issues – Retest.

## Task E-3-2: Evaluating System Testing

System testing ensures that all business functions and processes execute appropriately from the customer's view. Business processes are tested from beginning to end during system testing; this is sometimes referred to as end-to-end testing. The processes for performing system testing are described below:

- Create system test environment via a test Move to Production.
- Test inbound and outbound interfaces and related business processes.
- Test online business processes using relevant security (that is, user IDs, roles, and permission lists).
- Test batch business processes.
- Test reporting processes (SQR, PS/Query, nVision, and Crystal).
- Test customizations to business processes.
- Perform using test scripts.
- Compare expected results to actual results.
- Test – Document – Resolve issues – Retest – Document – Sign off.

## Task E-3-3: Evaluating Integration Testing

After system testing, you perform integration testing. In this stage, you test business processes and groups of related business processes within the application to determine that they function as designed. In addition, you ensure that any design flaws are resolved before user testing. The following list of activities describes integration testing:

- Create integration test environment via a test Move to Production.
- Test specific business processes.
- Test integration between modules and business processes.
- Perform using test scripts.
- Compare expected results to actual results.

- Test – Document – Resolve issues – Retest – Document – Sign off.

## Task E-3-4: Evaluating Parallel Testing

Parallel testing validates that the current production system and the upgraded database generate the expected results for specific business events. Parallel testing is optional, but frequently used to ensure that the new release will generate the same results given the same testing scenarios. The processes for performing parallel testing are described below:

- Create a parallel test environment via a test Move to Production.  
The Copy of Production should be taken before the major business processes/events are executed so that the same processes can be run during the parallel test.
- Retain any output from production processes for later comparison.
- Run the same business processes/events in the upgraded database.
- Compare results generated in the production system with the results generated using the upgraded database.
- Perform using test scripts.
- Test – Document – Resolve issues – Retest – Document – Sign off.

## Task E-3-5: Evaluating Performance Testing

You conduct performance testing to determine if the system can accomplish stated objectives within a specified time period. Performance of the current production system is often used as a baseline. The processes for performing performance testing are described below:

- Define performance objectives for each business process included in the scope of the test.
- Perform business process.
- Monitor performance.
- Compare actual performance and acceptance criteria.
- Perform using test scripts.
- Test – Document – Resolve issues – Retest – Document – Sign off.

## Task E-3-6: Evaluating User Acceptance Testing

User acceptance testing determines if day-to-day users can complete daily work activities within the system with an acceptable level of effort. For example, run through business processes such as hiring, terminating, and paying an employee in Human Resources or creating, editing, and posting journals in Financials. The processes for performing user acceptance testing are described below:

- Functional resources should execute test scenarios (with their appropriate production security access to ensure they have access to all the components, pages, and processes used in their daily functions).
- Perform using test scripts.
- User testing should not be performed with developer or Super User access.
- Test – Document – Resolve issues – Retest – Test – Document – Sign off.

## Task E-3-7: Evaluating Regression Testing

You perform regression, or re-testing, if problems were found and resolved or changes were made during any of the previous tests. This stage of testing validates the test Move to Production and Move to Production parts of the upgrade. When all the tests have received sign-off, you will use the initial Copy of Production to upgrade the production database/environment. You then perform a test move into the production environment and customers confirm that the test move executed successfully. At this point you conduct regression testing. The following tips will assist you with regression testing:

- Ensure that no new defects have been introduced during the move.
- Execute a predefined set of scripts to confirm the test move.
- Performed by Functional Resources before *Go Live*.
- Rerun scripts from previous testing.

---

## Task E-4: Determining the Testing Conditions

After you identify the types of testing to include in your upgrade, determine conditions for each stage of testing. Be sure to test the actual test Move to Production to resolve any technical issues in the upgrade process itself in addition to performing functional application testing. Perform the following actions for each testing type:

- Determine criteria for successful completion.
- Determine which tests you can run concurrently and which you must run serially.
- Set up test plans and test scripts you will need.
- Define the testing environment.
- Define issue resolution procedures.
- Define change control and migration procedures.
- Define which third-party tools you must install and configure.
- Identify database maintenance procedures, for example, backup and refresh.
- Evaluate the need for a testing tool to aid in the testing process.

---

## Task E-5: Developing Your Test Plan

If you have test plans from your implementation or previous upgrades, consider modifying them for this upgrade project, ensuring that you incorporate features and functions delivered with the new release. Use existing test plans and scripts wherever possible. Identify modifications during the fit/gap analysis and complete script generation during the initial upgrade. Based on the objectives and scope defined in your testing strategy, identify the following items for each type of testing:

- Test procedures
- Assumptions
- Timing
- Deliverables

- Acceptance criteria
- Roles and responsibilities
- Resource requirements
- Training requirements
- Test environment
- Data requirements
- Issue and change control tracking procedures
- Testing Tools

---

## Task E-6: Developing Test Scripts

The process of developing test scripts can assist with detecting problems in the requirements or design of an application. It requires thinking through the entire operation of the application. For this reason, you may find it useful to start preparing test scripts early in the upgrade cycle and, if possible, base them on existing test scripts from your implementation project or previous upgrade.

If you have test scripts from your original implementation, recycle them and modify them to accommodate new functionality. That way, you can be sure to cover your critical end-to-end business processes. You will also want to focus additional testing time on your customizations to verify that they have upgraded successfully.

If you do not have test scripts from your implementation, you can create them by documenting what you currently do within the system.

Create a test script for each business process to define the Action or Event, Input, and the expected result to determine if a feature of an application is working correctly. Functional people who are aware of current processes should write your test scripts. However, when writing test scripts, assume the person testing does not know how to use the system. Use the following procedure for developing test scripts:

- Test scripts should contain specifics, such as test identifier, test name, objective, test conditions and setup, input data requirements, steps, and expected results.
- Write as a step-by-step guide, stating what data should be entered, when, and where.
- Organize by module, business process, and process cycles.
- Create with full production security in mind.
- Create early in the upgrade process.

Make sure that your tests are consistent with the following tips:

- Action
  - Include the script name, description, and purpose.
  - Include the navigation steps within the PeopleSoft system.
  - Include navigation steps outside the PeopleSoft system.
- Input
  - Include security requirements: what User ID, Role, and Permission List should be used to perform the test.



- Specify key data elements: entering new or accessing existing data.
- Results
  - Include the exact results.
  - Print screens to support the results and print the report output.

---

## Task E-7: Reviewing Tips and Techniques

This section discusses:

- Reducing the Time of Upgrade Process
- Performing Security Testing
- Performing Testing on Up to Date Data
- Performing Test Move to Production
- Tracking Issues
- Reviewing Testing Tools
- Discussing Change Control
- Discussing Back Up Procedures
- Evaluating Unexpected Results
- Evaluating Reasons for Failure

### Task E-7-1: Reducing the Time of Upgrade Process

All testing can be performed at the end, including running the tests on the current system to obtain results for comparison. One way to reduce the overall timeframe of an upgrade is to execute the tests on the current system while the upgrade is in progress. This way you will have the results ready when the upgraded database is to be tested. This can be achieved by taking two copies of the production database at the start of the upgrade. Only one copy is upgraded, while the other remains at current release. The testing time is now reduced to only performing the tests on one database.

After each test Move to Production, you may want to turn over the upgraded database to the testing team while the technical team begins a new iteration of the test Move to Production. Conducting the two efforts in parallel may decrease the overall time required to upgrade. Any issues that are found by the testing team can be incorporated into the newly upgraded database as soon as it is available.

One approach that may be used for the Move to Production is to run the production database and upgraded database in parallel to ensure that key business processes operate as expected. Although this may require dual maintenance of data during the parallel testing period, it may minimize the impact of the actual Move to Production. If you are interested in using this method, once you have performed a test Move to Production to your satisfaction, schedule the production cutover weekend. Then, perform the next test Move to Production during that weekend. Bring the existing production system back up and run the upgraded database concurrently. If, after comparing the outcome of your critical business processes, you are satisfied with the results, simply set the upgraded database to your production system. If you are not satisfied with the results, make the necessary adjustments and perform another test Move to Production.

## Task E-7-2: Performing Security Testing

It is very important that you test security on each test pass you complete. Security is a vital part of the upgrade process and if it is not tested properly before you go live with your new release, there is a great potential for failure.

## Task E-7-3: Performing Testing on Up to Date Data

The previous technique will mean that testing is performed on *old* data. The copies of production may have been taken some weeks or months in the past. It will confirm that the data you started with has upgraded successfully. However, it may be required to perform the tests on the most recent data set as follows:

- Take a copy of production and upgrade
- Perform all phases of testing on the upgrade version up to unit testing
- Determine that the database is ready for full functionality testing

## Task E-7-4: Performing Test Move to Production

Performing a test Move to Production is a good technique for assuring database readiness as follows:

- Take two up to date copies of the production database.
- Perform the test Move to Production steps to upgrade one of the databases.
- Execute test scripts on the remaining database.
- Perform tests on the upgraded database and compare results.

## Task E-7-5: Tracking Issues

You should implement a method for tracking the tests and issues discovered during testing. Tracking issues and resolutions on a central document serves as a communication tool and minimizes duplication effort. The following tips should be considered while tracking issues:

- Categorize issues:  
Critical, Major, Minor, Cosmetic
- Use a central document repository or tracking tool

## Task E-7-6: Reviewing Testing Tools

SQA Robot :

- Records key strokes (like a macro recorder) into Visual Basic scripts.
- Useful for regression testing.
- SQA Manager—can be used to simulate multiple users.

Test Director by Mercury/Interactive:

- Tool that manages test scripts—tracks execution and defects.
- Useful for regression testing.

## **Task E-7-7: Discussing Change Control**

Make sure that you have a procedure for implementing changes during the testing stage. You may have several databases, if server space permits. All changes should be made in a master database. If an issue is found during testing, the resolution should be applied to the master database and promoted via pre-defined migration procedures. Remember that a master database will also be required to perform the Move to Production.

## **Task E-7-8: Discussing Back Up Procedures**

The following tips should be considered when backing up your data:

- Back up at baseline before testing (use a backup technique that will allow you to restore individual tables).
- Back up at key points for point in time testing.
- Implement refresh procedures to avoid duplicate data.

## **Task E-7-9: Evaluating Unexpected Results**

In the event you receive unexpected results and you cannot determine their cause, attempt to replicate any issues you encounter in your Copy of Production database on a delivered Demo database. If the issue does occur on Demo, it should be reported to the Global Support Center. Check PeopleSoft Customer Connection to see if a fix has been posted to resolve the issue.

See PeopleSoft Customer Connection (Updates + Fixes).

## **Task E-7-10: Evaluating Reasons for Failure**

The list below identifies reasons why your test plan might have failed:

- Testing strategy was poorly defined
- Test plans were poorly defined
- Test scripts were poorly defined
- Lack of resources and resource commitment
- Lack of understanding of the upgrade process



## APPENDIX F

# Preserving Queries and Tree Objects

This appendix discusses:

- Understanding Preserve Queries and Trees
- Preparing the Database
- Creating a New Project
- Comparing the New Project
- Copying the Project
- Testing the Project
- ReExporting the PeopleTools Tables

---

## Understanding Preserve Queries and Trees

This appendix contains information for preserving queries, trees, and tree structures. At the beginning of your upgrade, you should have informed your end-users and development team that your PeopleSoft system was frozen, meaning that no changes should have been made to any PeopleTools tables or objects including queries, trees, and tree structures. The freeze on PeopleTools changes is important because you will lose any changes to these objects made during an upgrade to PeopleTools tables. Occasionally, however, end-users may have to make critical changes to trees, tree structures, and PS/Query objects. If this has happened in your system, you can perform a process to preserve those additions and changes to trees, tree structures, and queries. You will have to work with your end-users and developers to obtain a list of queries, trees, and tree structures that you need to preserve.

You will run through the test Move to Production (MTP) steps several times for practice and testing purposes. Please note that you have the option to perform the preserving queries and trees procedure during each of your test Move to Production runs, but you must perform it during the last run of the test Move to Production. If you do not perform this procedure during your last run to preserve the trees, tree structures, and queries that have been changed since the beginning of your upgrade, they will be lost.

---

**Note.** The process outlined in this appendix to preserve trees and queries should be performed prior to data conversion so that any additional conversion would be taken care of by the appropriate data conversion programs.

This appendix includes instructions to prepare your database and create a project on which to preserve your queries, trees, and tree structure changes.

---

---

## Task F-1: Preparing the Database

In this step, you create a new copy of your current production database, perform steps on the new copy, and run scripts against the new copy to update the release level.

To prepare the database:

1. At the beginning of the test Move to Production, you should make a new copy of your current production database. To preserve queries and trees, you need to make not only that Copy of Production but also an additional copy of your current production database. For clarity, PeopleSoft refers to this additional copy of your production database as the Tree/Query Copy of Production database. So now you should have a Copy of Production database and a Tree/Query Copy of Production database.
2. Perform the test Move to Production on your Copy of Production database.
3. To obtain the queries and trees that you want to preserve, the Tree/Query Copy of Production database needs to be at the same release level as the Copy of Production database on which you just completed the test Move to Production. To update your Tree/Query Copy of Production to the same release, you run release scripts against this database. PeopleSoft refers to this as “reling up” the database. Use the Custom Compare template to “rel up” your database. Select the Product Line *PEOPLETOOLS* when configuring your Change Assistant job.

---

## Task F-2: Creating a New Project

Now that your Tree/Query Copy of Production is at the same release as your Copy of Production database, you create a project in the Tree/Query Copy of Production that contains all of the queries and trees that you wish to preserve.

To create a new project:

1. Using PeopleTools 8.4x, sign on to the Tree/Query Copy of Production using a valid PeopleSoft User ID and launch Application Designer.
2. Select File, New...
3. Select *Project* for Object Type.
4. Select File, Save Project and enter a project name; for example, *PRESERVED*.
5. Select the Upgrade tab in Application Designer.

---

**Note.** Queries and trees do not appear in projects under the Development tab in Application Designer. To see the queries and trees that you will insert into the *PRESERVED* project in the next step, you must make sure that you are using the Upgrade view of Application Designer.

---

6. Select Insert, Definitions into Project...
7. Select Queries from the Definition Type drop-down list box and click Insert.
8. Using your list of identified queries that need to be preserved, highlight each one of those queries from the Application Designer list.

You can highlight more than one by holding down the Control (CTRL) key while you click the name of the query.

9. After you have highlighted all of the queries that you want to preserve, click Insert, then click Close.  
Under the PRESERVED project name in the Upgrade view of Application Designer, you will see Queries as an object type in the project.
10. Double-click on queries under the PRESERVED project to see a listing of all of the queries to preserve in the right-hand window of Application Designer.
11. Select File, Save Project.
12. Repeat steps 6 through 11 for trees and tree structures.  
Now your PRESERVED project should contain all of the queries, trees, and tree structures that you want to preserve.

---

## Task F-3: Comparing the New Project

In this step, you compare the queries, trees, and tree structures that are in your PRESERVED project against your Copy of Production database. Because the tree objects in your PRESERVED project are not comparable objects in Application Designer, you must manually compare the tree objects that you want to preserve. During the query and tree structure compare process, the Application Upgrade utility sets the project flags. These flags determine whether the following actions will occur:

- Changes will be performed on the Copy of Production (Target) database when you perform the export and copy.
- Changes will be tagged as *Copy* or *Delete* operations.
- The project flags will be set to automatically take these actions or not.

These settings are determined based on whether or not the objects in the project currently exist on the Copy of Production (Target) database.

To compare the new project:

1. Using PeopleTools 8.4x, sign on to the Tree/Query Copy of Production using a valid PeopleSoft User ID and launch Application Designer.
2. Select File, Open...
3. For Definition, select Project and click Open to display the list of projects.
4. Select the PRESERVED project and click Open.
5. Select Tools, Compare and Report.
6. Sign on to your Copy of Production.
7. From the Object Type box, select *Queries and Tree Structures*.
8. Click Options...
9. Select *PeopleSoft Vanilla* for the Target Orientation.
10. Select *Project* for the Compare Type.
11. Verify that the Compare Report output directory is set to the correct location.
12. Select the Report Filter tab and set the report filter check boxes appropriately for your compare.
13. Click OK.

14. Select Compare.
15. Review the compare reports for queries and tree structures. In addition, perform a manual compare of the trees that you want to preserve. Based on the results of this review, set the Action and Upgrade check box appropriately in the PRESERVED project.

---

## Task F-4: Copying the Project

In the following steps, you copy the PRESERVED project to the Target database. This is the Copy of Production database on which you ran the test Move to Production.

To copy the project:

1. Using PeopleTools 8.4x, sign on to the Tree/Query Copy of Production using a valid PeopleSoft User ID and launch Application Designer.
2. Select File, Open...
3. For Definition, select *Project* and click Open to display the list of projects.
4. Select the PRESERVED project and click Open.
5. Select Tools, Upgrade, Copy.
6. Sign on to your Copy of Production database.
7. Make sure that the Reset Done Flags and Copy Project check boxes are selected.
8. Click Select All.
9. Click Copy.
10. Using the Upgrade view of the PRESERVED project in Application Designer, review the Done flags in the project to make sure that all of the objects that you wanted to preserve were copied to the Target database.

---

## Task F-5: Testing the Project

Now that the queries, trees, and tree structures that you wanted to preserve are in the Copy of Production database, you must test and re-test and make any necessary changes if the test results are not what you expected.

---

## Task F-6: ReExporting the PeopleTools Tables

Once you are satisfied with the test results, you must re-export the PeopleTools tables to actually preserve the queries, trees, and tree structures. During your test Move to Production, you ran MVPRDEXP.DMS to export the PeopleTools tables. You will use the output files created from running this job as input files during your Move to Production. Because these files were created before copying the queries, trees, and tree structures that you wanted to preserve, the files do not contain the preserved objects, so you must run the MVPRDEXP.DMS script again. Running the MVPRDEXP.DMS script again ensures that you have the most current PeopleTools tables.

To re-export the PeopleTools tables:



1. As a PeopleSoft user, launch Data Mover against your Copy of Production database and run the following script:

`\PS_HOME\SCRIPTS\MVPRDEXP.DMS`

2. Use the output files created during your final Move to Production.



## APPENDIX G

# Reviewing Batch Program Changes

This appendix discusses:

- Reviewing Batch Program Changes

---

## Reviewing Batch Program Changes

Some of the batch processes have been rewritten from the previous version. Some were converted to Application Engine, while others were rewritten and renamed. The following table lists the batch programs (COBOL and SQR) that have been changed. Use this list when you review your batch process customization. If you know you have customized a particular program on this list, review the new program to verify whether you still need that customization.

Program Name in Previous Release	New Release Program Environment	New Release Program Name
BEN101.SQR	Removed as obsolete, replaced by plan-type-specific model SQRs.	Replaced by BEN102.SQR, BEN103.SQR, BEN104.SQR, and BEN105.SQR.
BEN200.SQR BEN201.SQR BEN202.SQR	Not Applicable: Removed as obsolete.	Not Applicable: Removed as obsolete.
BEN500.SQR	SQR Program re-delivered as an Application Engine.	BN_BEN500_AE
BEN100.SQR	Replaced by BEN100A.SQR and removed as obsolete.	Replaced by BEN100A.SQR and removed as obsolete.
BEN110.SQR	Split into two processes: BEN110 Application Engine "Snapshot Premium - Calculation" and BEN110.SQR "Snapshot Premium - Report."	Now invoked through Processing Job BEN110_J "Snapshot Premium - Reporting."
BEN702.SQR (Flat) BEN704.SQR (Age) BEN714.SQR (Service) BEN731.SQR (Salary)	Individual Rate Table reports were removed and consolidated into a new report.	Replaced by BEN741.SQR (Benefits Rate Table).



## APPENDIX H

# Reviewing Benefits Changes

This appendix discusses:

- Reviewing Consolidation of FSA Balances
- Understanding Changes to Snapshot Tables
- Reviewing Savings Management Under EGTRRA of 2001
- Reviewing COBRA Administration
- Reviewing Benefit System Indicator on Job
- Reviewing Benefit Record Number Display
- Reviewing Effective Dating of Dependents
- Reviewing Effective Dating of Company Cars
- Understanding Benefit Program Enhancement
- Reviewing Changes to Relationship Usage
- Reviewing Spending Account Changes
- Reviewing Consolidation of Benefit Rates
- Reviewing Consolidation of Coverage Calculations

---

### Task H-1: Reviewing Consolidation of FSA Balances

FSA Administration maintains year-to-date balances on the FSA Benefit table (the same table that manages an employee's enrollment). This design leads to complications, most noticeably in a Benefits Administration environment. If the event that generated the enrollment is reprocessed, any FSA Administration balance activity is lost. It is then possible to over-reimburse on an FSA claim. A further complication is that FSA claims information should really be managed at an *employee* level (EMPLID) and not at a *job* (EMPL\_RCD) level, since FSA is a regulated plan type with mandated maximum benefits.

Therefore, we are moving the maintenance of year-to-date FSA balances onto a new table, PS\_FSA\_SUMMARY, which is keyed only by Employee ID, Plan Type, and Balance Year. This new structure requires a consolidation of balance information across possibly multiple enrollment records (in a multiple-jobs environment). The system will attempt during upgrade to create a single FSA Summary entry for each employee for each FSA Plan and each enrollment year found. Because this is a significant design change and because PeopleSoft recognizes that clients may have developed custom interfaces or reports against the existing data, the balance information currently on the PS\_FSA\_BENEFIT table will not be removed during this release, although it will no longer be actively maintained. Clients should view this data as obsolete, however, and assume that these fields will be dropped in a subsequent major release.

---

## Task H-2: Understanding Changes to Snapshot Tables

The Carrier Interface Load program BEN100.SQR has been replaced by a more efficient process, BEN100A.SQR. The “snapshot” tables used by the old process were PS\_BEN\_PERS\_DATA and PS\_BEN\_PLAN\_DATA, and were the basis for the Carrier Report/Interface BEN101.SQR. The new process uses a more complete set of snapshot tables, PS\_BN\_SNAP\_PER, PS\_BN\_SNAP\_PLAN, and PS\_BN\_SNAP\_JOB. Because these are point-in-time snapshots, no conversion is performed (you would simply run a new snapshot). A new configurable loading process and more complete reporting and interface samples are delivered with this release. Note that since the new Enrollment Snapshot process was originally introduced as a posting against release 8 SP1, sufficient time has elapsed for customers to migrate to the new process. Therefore, PeopleSoft is now dropping the obsolete program and support tables

---

## Task H-3: Reviewing Savings Management Under EGTRRA of 2001

The Economic Growth and Tax Relief Reconciliation Act of 2001 changed the manner in which certain regulatory limits are applied to savings contributions. Prior to the Act, the administration of 403(b) plans required the maintenance of life-to-date earnings and contribution information, and also required that projections be made through year-end to anticipate limits. Because of this, the Savings Management record SAVINGS\_403B\_EE was keyed by Savings Plan Type and included a variety of fields for managing projected data. Under the Act, the focus is now more on managing voluntary extensions to specific regulatory limit types, such as 402(g); therefore, not only are the projection fields being dropped, but the record’s key structure has been changed and the table renamed Savings Extensions and Overrides.

Limit Types/Exception Reasons have been expanded to include “Limit 401(a)/Adjust Eligible Earnings YTD,” which allows the user to adjust eligible earnings used in 401(a) limit processing. The second Limit type/Exception reason is the “Hardship Withdrawal 402(g) limit reduction,” which was taken off of the 402(g) Catch up row. The corresponding Hardship Withdrawal Date, which regulates the Savings Plan suspension period after a hardship withdrawal, has been moved to a new Enrollment Suspension grid. Note that Exception Reason is more descriptive of how it affects the calculation; for example, Extend Limit, Reduce Limit, or Adjust Eligible Earnings YTD.

To preserve the data which is currently recorded in the Savings Management record, we retain the table SAVINGS\_403B\_EE in the system as a deprecated object, and make this table available through PS Query for reference via the Employee Benefit Enrollments (EE BENEFIT INFO) access group of the Base Benefits query access tree. All current and future management is performed within the new SAVINGS\_MGT\_EE table, using essentially the same pages and processes. Note that the SAVINGS\_403B\_EE table is considered obsolete, will not be further maintained by PeopleSoft, and is subject to being removed from the system in a subsequent major release. Further upgrade is performed for all rows in Savings Management with the Limit Type/Exception Reason “402(g) Catch-up” and an existing Hardship Withdrawal Date. Review the new Limit Exception “402(g) Hardship” and Enrollment Suspension rows generated for accuracy and completeness.

---

## Task H-4: Reviewing COBRA Administration

PeopleSoft created a table (PS\_BEN\_CONTACT\_TBL) for storing names and address information for internal benefits administrative contacts. This table supports HIPAA certification and COBRA Administration notices. The system converts existing COBRA Administrator records into the new general contact format, and then updates associated Benefit Programs to use the new corresponding administrative contact. The conversion of existing COBRA Administrator records is problematic because the same contact person may administer COBRA for many Benefit Programs. Under the old structure, many identical COBRA Administrator records were created (one for each Benefit Program). The new system identifies and consolidates these multiple records during conversion by grouping them by the administrator NAME field. However, if the NAME of the Administrator differs on multiple records, the system creates several new contact records for them in PS\_BEN\_CONTACT\_TBL. If you are using PeopleSoft COBRA Administration, review the results of the conversion by selecting Set Up HRMS, Product Related, Base Benefits, Plans and Providers, Administrative Contacts.

---

## Task H-5: Reviewing Benefit System Indicator on Job

In the new PeopleSoft release a new value was added to the translate list for Benefit System “OT” (Other or Not managed by PeopleSoft Benefits). This complements the existing values of the “BN” (Base Benefits) and “BA” (Automated Benefits Administration). This new value, in conjunction with changes to the Person Model, allow for better management of contingent workers and non-employee persons-of-interest. These persons can be assigned to Benefits System OT and will be screened out of most Benefits-related search views and processes. Persons assigned to Benefit System OT will not have default Benefit Program assigned to them in the Job Data pages.

During the upgrade, we attempt to identify persons who are assigned to Benefit System “BN” because no other choice was available. Most notably, these are persons for whom prior hiring logic did not require a Benefits Program Participation record. All persons lacking a Benefits Program Participation record (PS\_BEN\_PROG\_PARTIC) are moved to the Benefit System OT. This can leave many persons assigned to Benefit System BN for whom you don’t actually manage benefits. Unfortunately, the upgrade process cannot reliably identify such persons because of the wide range of business practices in use. (For example, some companies want certain persons assigned to a Benefit Program for administrative or third-party purposes, even though these persons never have active benefit elections recorded for them). Leaving such persons in Benefit System BN will not be a problem, as the system will continue to process (or ignore) them as before.

---

## Task H-6: Reviewing Benefit Record Number Display

In some prior PeopleSoft releases, a configuration parameter on the Installation Table was used to indicate whether multiple simultaneous jobs for employees were managed. However, this system capability is now permanently turned on. (This tracks persons-of-interest and contingent workers, and allows these persons to transition to employees while retaining their original high-level Person ID.) Because of this system-wide change, the concept of the Benefit Record Number is now more prominent than it was in previous releases. In fact, all Base Benefits election pages now display Benefit Record Number (BENEFIT\_RCD\_NBR) rather than Employment Record Number (EMPL\_RCD).

In the past, if you managed only single jobs for your employees, you were “shielded” from the Benefit Record Number. This was possible because the employee’s single Benefit Record Number was generated to equal the employee’s single Employment Record Number, so the concept of Benefit Record Number was kept “hidden” for clarity. If you manage multiple jobs for your employees, you know that Benefit Record Number is the key to grouping together several jobs that share the same set benefits. For these customers, the Employment Record Number displayed in the Base Benefits corresponded to the “primary” job within the Benefit Record Number group.

Going forward, the Base Benefits search dialogs and pages use and display Benefit Record Number.

---

**Note.** This does not change how benefits are managed, and is not a functional change in how the PeopleSoft system behaves.

---

This is simply an attempt to make the system visually and functionally consistent. If you manage only single jobs for your employees, you can ignore the Benefit Record Number, which is automatically assigned to employees, contingent workers, and persons-of-interest in a manner consistent with previous releases. If you manage multiple jobs with multiple benefit tracks, the new search view criteria and page displays result in improved usability.

---

## Task H-7: Reviewing Effective Dating of Dependents

Dependent/Beneficiary information is now effective-dated. The existing dependent information has been restructured in a manner similar to the core Person information: a core set of identifying information that does not change with time (PS\_DEP\_BEN); biographical information that is time sensitive (PS\_DEP\_BEN\_EFF); and separate records for Name (PS\_DEP\_BEN\_NAME), Addresses (PS\_DEP\_BEN\_ADDR), and National ID (PS\_DEP\_BENEF\_NID). Most processes that use dependent (or beneficiary) information will be fully effective-date enabled. They will retrieve or process dependent information based on the state of that data on the historical or functional event date or “as-of” date applicable to the benefits process being run.

---

## Task H-8: Reviewing Effective Dating of Company Cars

Company car data is now effective dated with effective status. In previous PeopleSoft releases, you may have directly deleted the cars (CAR\_DATA) from the table when they were no longer required to avoid incorrect allocation. In the new PeopleSoft release, you can make the cars active or inactive and maintain history.

---

## Task H-9: Understanding Benefit Program Enhancement

Customers have asked for an “empty” Benefit Program, one with no Plans associated with it, for use in various administrative situations. For example, PeopleSoft Payroll for North America expects a payee to be assigned to a Benefit Program even if that individual is not being administered through Base Benefits. Assigning this person to an “empty” Program emphasizes the administrative nature of the assignment and prevents unintentional enrollments from being entered. To facilitate this, a new manual Benefit Program called PSX with an effective date of 1900-01-01 is being delivered as system data. (This is because the business validations performed by the Benefit Program Definition component prevent the user from directly creating such an “empty” program themselves.)



---

## Task H-10: Reviewing Changes to Relationship Usage

This section discusses:

- Understanding Changes to Relationship Usage
- Reviewing Cobra Event Rules
- Reviewing Dependent Relationship Rules
- Reviewing HIPAA Changes

### Task H-10-1: Understanding Changes to Relationship Usage

For consistency in the use of Relationship codes, we are consolidating any relationships codes in the Translate Table that currently reflect gender into a single genderless code. For example, *Son* and *Daughter* are being consolidated into *Child*, since in general gender is available as a separately tracked personal attribute. All tables that contain the field Relationship will undergo data conversion to reflect the non-gender translate values.

### Task H-10-2: Reviewing Cobra Event Rules

Cobra Event Rules (CBR\_EVENT\_BENEF) is being modified to use Covered Person Type instead of Relationships, in order to streamline and standardize the functionality with the existing functionality in the Benefits Administration product. You should validate the content of this record after upgrading, since the conversion of duplicate genderless Relationship rows might have impacted your production data.

### Task H-10-3: Reviewing Dependent Relationship Rules

The Dependents Relationship Table (DEPBEN\_TYPE\_TBL) is specifically upgraded to convert to genderless relationship codes. To eliminate duplicates, only the most current rows for any grouping of old gender-specific translate values was converted. Customized rows that do not conflict with the new PeopleSoft-maintained non-gender values are converted, but overlapping rows are deleted. You should validate the content of this record after upgrading.

### Task H-10-4: Reviewing HIPAA Changes

The EDI mapping table (BN\_834\_MAP\_TBL) is specifically upgraded to convert to genderless relationship codes. Customized rows that do not conflict with the new PeopleSoft-maintained non-gender values are converted, but overlapping rows are deleted. You should validate the content after upgrading.

---

## Task H-11: Reviewing Spending Account Changes

This section discusses:

- Understanding Spending Account Setup
- Reviewing Changes to Spending Account Pledge Limits
- Reviewing FSA Claims Processing Setup

## Task H-11-1: Understanding Spending Account Setup

A new field, Spending Account Type (BN\_SPEND\_ACCT\_TYPE), has been added to the FSA Benefit Table. It is used to distinguish whether the Spending Account being created is a traditional Flexible Spending Account (which assumes claims management) or a newly supported Health Savings Account. The default value for this new field is *Flexible Spending Account*. During the Record-Alter step of applying the upgrade, all existing rows will receive this default value, since that is the only type of Spending Account that was previously supported. Further, for plan types 60, 61, 65, and 66, this attribute cannot be changed. This is because certain assumed functional and regional behaviors for these standard FSA plans are built into the code. For all other plan types, the Spending Account Type is open for assignment, although all benefit plans under a given plan type *must* have the same Spending Account Type. Note also that the system still has limitations about its ability to manage client-defined plan types within the 6X series.

A new field, Plan Contribution (PLAN\_CONTRIBUTION), has also been added to the table. This concept is borrowed from Savings Plans and controls the requirements for employee contributions (Annual Pledge) when enrolled in a Spending Account. That is, once enrolled, this flag controls whether an employee contribution is required, is optional (perhaps because the employer is also contributing), or not allowed (as might be the case for a fully employer-paid HSA plan). To retain existing system behavior, the default for this field during the Record-Alter step is *Contribution Required*.

## Task H-11-2: Reviewing Changes to Spending Account Pledge Limits

For Spending Accounts (6X plans) the Minimum and Maximum Employee Annual Contribution limits have been moved from the Benefit Program definition (from the Plan-level on PS\_BEN\_DEFN\_PLAN) onto the Spending Account plan table (PS\_FSA\_BENEF\_TBL). This provides better control over plan-specific limits. The data conversion performed on existing limit data is consequently a “best guess.” This is because the same 6X plan may appear in more than one Benefit Program, but with different plan-type-level limits. Thus, the existing limit definitions cannot be accurately mapped from their current plan-type level to the new individual benefit-plan level. The data conversion process will copy the lowest non-zero Minimum limit and the highest non-zero Maximum limit found for each Benefit Plan to the Spending Account Plan Table. As part of your implementation of this upgrade release, you need to review all of your existing 6X Spending Accounts and verify the limit information for them. Note that the default numeric value of 0 for either limit will be interpreted as unlimited (or limit not used). Note that in order to reduce the impact of this feature, the existing limit fields are not physically removed from the PS\_BEN\_DEFN\_PLAN table, but rather simply hidden on all application pages, set to zero in the table, and ignored by all processing.

## Task H-11-3: Reviewing FSA Claims Processing Setup

An FSA Admin Configuration tab has been added to the FSA Benefits Table setup to capture the information needed to properly implement the optional grace period, where overlapping periods exist for different calendar years and claims may be paid from either available remaining pledge. This information is only needed if FSA claims will be processed using the FSA Administration product.

Two new date fields, Service Date From and Through, are used to determine the service dates that are valid for claims submitted. The following online restrictions apply:

- Claims must have a service date within the service date range to be paid under the plan.
- The beginning service date will default to the effective date of the plan; for example, 1/1/2006.
- The end service date will default to 12 months after the effective date of the plan; for example, 12/31/2006.
- The end service date must be greater than or equal to the beginning date.

- The service date range may be longer than 12 months and may overlap with service dates of other benefit plans that the employer offers in which the employee is enrolled.

A new Plan Year field has been added, in order to properly close only the relevant plans. All plans will need to be identified with a Plan Year value. This field has a prompt from which valid years, such as 2005 or 2006, can be selected. This value is used to group plans for use in the FSA Closure process. A new field is added to the FSA Closure run control, from which the plan year is selected. Only those plans with a matching Plan Year Name value will be displayed and then can be selected for closure.

The minimum check amount for FSA plans, which was previously determined at the benefit program level, has been moved to the benefit plan level. This allows FSA plans within one benefit program to have different minimum check amounts. It is used to determine the minimum check amount when generating payments for the plan in the same manner as before.

The upgrade process will populate the new fields as follows:

- Service dates from and through will default to the beginning and the end of the current year when the upgrade is run for all combinations of FSA plan types and benefit plans where the effective date is before the current date. In cases where future effective dates exist, a 12-month period will be assigned using the effective date as the starting service date.
- The plan year will be the current year, or the year of the future effective date, depending on the case.

Be aware that the stated assumptions might not coincide with your implemented functionality, and values will need to be reviewed after conversion by selecting Set Up HRMS, Product Related, Base Benefits, Plan Attributes, FSA Benefits Table. Also, since each effective dated row covers only a specific service period, you will need to introduce a new effective dated row for each plan type and benefit plan each year going forward in the FSA Benefits table. This is only required if FSA claims will be processed using the FSA Administration product.

---

## Task H-12: Reviewing Consolidation of Benefit Rates

In this release, we are consolidating all benefit rate data into a single common structure. The individual Age-Graded, Flat, Salary-Percent and Service tables will be dropped, and all rate data migrated to the new Benefit Rate Table. This table has a flexible architecture that can support all rate types, and will allow for the definition of user-defined rate types as well. The data conversion process has several points that require review:

- **Rate Table IDs:** Because the current benefit rates exist in several individual tables, the possibility exists that the Rate Table IDs defined on those tables could conflict (be duplicates) when consolidated. To manage this, each Rate Table ID will be checked as it is converted, and if a conflict is detected, the Rate Table ID will be modified to make it unique. Appending a Rate Type code to the end of the Rate Table ID does this. In prior releases, the Rate Type field was controlled by a set of Translate Table values. The existing values are Age-Graded (rate type=1), Flat (rate type=2), Salary-Percentage (rate type=3), and Service (rate type=4). So as an example, if a Service-rate Rate Table ID “ABC” was found to conflict with an ID that had already been converted, it would be renamed as “ABC-4”. All references to the old Rate Table ID in Benefit Program definitions will be updated. You can review the set of Rate Table IDs that experienced conflicts by using the search criteria “%-” in the new Benefit Rate Table component (Navigation: Set Up HRMS, Product Related, Base Benefits, Rates and Rules, Benefit Rates.)
- **Zero-base on Numeric Ranges:** The Age-Graded table and the Service table both contain ranges used to determine the correct rate. In prior releases, the system allowed two situations that make interpretation difficult. First, some Rate Tables do not have a lowest range defined. For example, the first entry may be 5 to 15. Thus the range 0 to 4 is not managed. (Fortunately, the system would default to a zero rate for values in these undefined ranges.) During data conversion, the system will insert an “anchor” value of

zero for all ranges and set the associated rates to \$0.00. Without a specific lower functional limit, the new benefit rate architecture will return an error if a value falls outside the defined table ranges. Secondly (and more commonly), the first range in the rate table may start with 1 rather than 0. For example, the first age range might be 1-20, which technically omits newborns. (Again, the system previously would default to a zero rate for ages below 1). During data conversion, the system will detect that a range starts at 1 and automatically set it to start at zero instead. Since most values (age and service) can functionally be zero in some situations, the new benefit rate architecture will return an error if a value falls outside the defined table ranges. You may want to review your converted Age-Graded and Service rates to confirm that these actions have been made appropriately. (Navigation: Set Up HRMS, Product Related, Base Benefits, Rates and Rules, Benefit Rates.)

- **Currency Code:** Benefit Program definitions currently display a Currency Code as an informational item. However, neither PeopleSoft Benefits nor PeopleSoft Payroll for North America perform any type of currency conversion. In previous releases, benefit rates did not include a currency code. Thus, the burden was on the user when setting up a Benefit Program to only use rates that matched the Program's currency; the system was unaware of any currency mixing. For HCM 9, the Benefit Rate Table now includes a Currency Code as an informational item (except for Percent-of-Salary rates, which have no implied currency). When setting up a Benefit Program, the system will validate that a Rate's currency matches the Program's currency to reduce setup errors. During data conversion (benefit rate consolidation), the system will attempt to assign an appropriate currency code to each Rate Table ID, based on its usage in Benefit Programs. There are two considerations here that should be reviewed: Rates may be defined that have never been used in a Benefit Program, or Rates may exist that have been used in several different Benefit Programs whose currency codes are not consistent. In the first instance, the converted Benefit Rate will have a blank Currency Code. This will not affect its use (the system will treat that as a "wildcard" and allow it to be used with any Program currency) but it should be corrected as soon as possible for the sake of simple data integrity. In the second instance, during data conversion the system will simply assign the currency code found on the alphabetically lowest Benefit Program using that Rate Table ID. This should be reviewed for correctness. (Select Set Up HRMS, Product Related, Base Benefits, Rates and Rules, Benefit Rates.)

---

## Task H-13: Reviewing Consolidation of Coverage Calculations

In prior releases, elements of *coverage* existed not only on the Benefit Plan attribute tables, the Life ADD Table and the Disability Plan Table, but also on the Calculation Rule Table. The Life ADD Table contained the coverage formula components (Factor X Salary + Flat Amount), the Disability Plan Table contained the Maximum Monthly Benefit and an implicit formula of Salary / 12, and the Calculation Rules Table contained the Coverage and Premium As-Of date rules, the Benefit Base to use, Multiple-job combination rules, Rounding rules, and Coverage Minimum/Maximum. For this release, we consolidated these components into a "Coverage Formula."

During data conversion, the system analyzes the usage of Life and Disability benefit plans and Calculation Rule IDs in Benefit Programs. Based on the combinations found, the system creates new Coverage Formulas to duplicate the coverage calculations performed in previous releases. The number of combinations could be quite large, and many of them could be unintentional (or even duplicates) since the Benefit Program allows complete flexibility in associating calculation rules with benefit plans, and since multiple benefit plans may exist with the same attributes but from different vendors or with different descriptions. To reduce the number of formulas created, and to avoid perpetuating unintentional variations, we restrict the data conversion; for each distinct set of Life or Disability attributes, we combine the set with only the single alphabetically lowest Calculation Rule ID with which it is associated. We then create a *model* Coverage Formula based on that combination, and update all relevant Life and Disability plans to reference the new Coverage Formula ID. In doing this, we assume that the same Life or Disability plan is not being used with a wide variety of calculation rules (different rounding rules, different Minimum/maximum, etc.), in order to avoid creating a large number of Coverage Formulas. The concept of the reusable Coverage Formula is significant enough to warrant performing only this *model* conversion, and requires a small re-implementation of your Life and Disability plans. This should be as simple as reviewing the data conversion results and confirming the actual coverage calculations in test deduction calculation runs. (Navigation: Set Up HRMS, Product Related, Base Benefits, Rates and Rules, Coverage Formula.)

To facilitate the use of the automatically converted Coverage Formulas, we have implemented a naming convention for Coverage Formula IDs that is representative of the formula's content, 99|X|X|999X|X|X|, which is described in the following table.

Byte Position	Format	Content
1-2	99	Salary factor rounded to an integer for Life, DS for Disability
3	X	The Benefit Base used, coded as Job Compensation Rate (C) or Annual Benefits Base Rate (A).
4	X	Indication of whether the Base combines across multiple jobs, coded as Multiple (M) or Single (S).
5-8	999X	The Flat Amount for Life, if any, added to the coverage calculation, or Maximum Monthly Benefit for Disability, in a form of 500, 250K, or 1.5M, as appropriate.
9	X	The Coverage Base As-Of date rule, coded as Check Date (C), Pay End Date (P), Last Year On MM/DD (L), This Year On MM/DD (T)
10	X	A letter used to identify, in sequence, otherwise identical formulas (A, B, C, D, ...)

For example, Coverage Formula ID “02CS050KCA” would represent a Life coverage formula of “2 x Salary + 50,000,” where the *Salary* is the Compensation Rate from the primary Job (not combined for multiple jobs) as determined on the Check Date. The final *A* indicates that this is the first such formula created. Note that the Rounding rules, if any, are not represented in the Coverage Formula ID string, for lack of space.



# APPENDIX I

## Reviewing HRMS Changes

This appendix discusses:

- Reviewing Record and Field Conversions
- Reviewing Records with Duplicate Key Structures
- Understanding Applicant Data Normalization
- Reviewing the National ID Expiration Record
- Reviewing Schedule Defaults

---

### Task I-1: Reviewing Record and Field Conversions

This section discusses:

- Understanding Record and Field Conversions
- Status Conversions
- Reviewing Profile Management Table Deletions

#### Understanding Record and Field Conversions

This section discusses a number of columns in your database that will be converted during the upgrade but will need some functional analysis performed prior to running the data conversion scripts. Review the items below to determine if you want to modify the delivered scripts prior to running them.

#### Task I-1-1: Status Conversions

In the new release, the status indicators for six Recruit Workforce Records are changed from Translate Values to system-defined status codes. The following record.fields are converted:

- JOB\_REQUISITION.JOB\_REQ\_STATUS
- APPLICATN\_DATA.APP\_STATUS
- POSN\_APPLIEDFOR.APP\_STATUS
- OFFER.RESPONSE\_OFFER
- APPLICATN\_ROUTE.RESPONSE\_CD
- INTERVIEW.RECOMMENDATION

In the task, “Making Functional Decisions,” you are prompted to add translate value to status code mappings for translate values that you have added to the system. Below is a list of the Status Codes that

are delivered in the System Data for ER\_STATUS\_TBL and ER\_STS\_RSN\_TBL. You should map the additional translate values to these status codes here and you should map the new translate values to status codes that you will later add to the system after the upgrade is complete.

- ER\_STATUS\_TBL

Recruitment Area	Status	Code Description
1-JOB REQUISITION	010	010-Open
1-JOB REQUISITION	020	020-Applied
1-JOB REQUISITION	030	030-Preliminary screen
1-JOB REQUISITION	040	040-Final screen
1-JOB REQUISITION	050	050-Route
1-JOB REQUISITION	060	060-Interview
1-JOB REQUISITION	070	070-Offer
1-JOB REQUISITION	080	080-Ready to hire
1-JOB REQUISITION	090	090-Re-Opened
1-JOB REQUISITION	100	100-Hold
1-JOB REQUISITION	110	110-Filled/Closed
1-JOB REQUISITION	120	120-Cancelled
2-APPLICANT	010	010-Active
2-APPLICANT	020	020-Hired
2-APPLICANT	030	030-Inactive
3-POSITION APPLIED FOR (DISPOSITION)	020	020-Applied
3-POSITION APPLIED FOR (DISPOSITION)	030	030-Preliminary screen
3-POSITION APPLIED FOR (DISPOSITION)	040	040-Final screen
3-POSITION APPLIED FOR (DISPOSITION)	050	050-Route
3-POSITION APPLIED FOR (DISPOSITION)	060	060-Interview
3-POSITION APPLIED FOR (DISPOSITION)	069	069-Preliminary offer decided



Recruitment Area	Status	Code Description
3-POSITION APPLIED FOR (DISPOSITION)	070	070-Offer
3-POSITION APPLIED FOR (DISPOSITION)	073	073-Request wrks cncl approval
3-POSITION APPLIED FOR (DISPOSITION)	075	075-Preliminary offer notified
3-POSITION APPLIED FOR (DISPOSITION)	076	076-Preliminary offer accepted
3-POSITION APPLIED FOR (DISPOSITION)	077	077-Preliminary offer rejected
3-POSITION APPLIED FOR (DISPOSITION)	078	078-Hire decided
3-POSITION APPLIED FOR (DISPOSITION)	080	080-Ready to hire
3-POSITION APPLIED FOR (DISPOSITION)	090	090-Hired
3-POSITION APPLIED FOR (DISPOSITION)	100	100-Hold
3-POSITION APPLIED FOR (DISPOSITION)	110	110-Reject
3-POSITION APPLIED FOR (DISPOSITION)	120	120-Withdrawn
3-POSITION APPLIED FOR (DISPOSITION)	130	130-Selected
3-POSITION APPLIED FOR (DISPOSITION)	140	140-Inactive
4-ROUTE (RECOMMENDATION)	010	010-Routed
4-ROUTE (RECOMMENDATION)	020	020-Invite for interview
4-ROUTE (RECOMMENDATION)	100	100-Hold
4-ROUTE (RECOMMENDATION)	110	110-Reject
4-ROUTE (RECOMMENDATION)	120	120-Withdrawn

Recruitment Area	Status	Code Description
5-INTERVIEW (RECOMMENDATION)	010	010-Schedule interview
5-INTERVIEW (RECOMMENDATION)	020	020-Make offer
5-INTERVIEW (RECOMMENDATION)	030	030-Interview further
5-INTERVIEW (RECOMMENDATION)	100	100-Hold
5-INTERVIEW (RECOMMENDATION)	110	110-Reject
5-INTERVIEW (RECOMMENDATION)	120	120-Withdrawn
6-OFFER (RESPONSE)	010	010-Extend
6-OFFER (RESPONSE)	020	020-Accept
6-OFFER (RESPONSE)	030	030-Acceptance withdrawn
6-OFFER (RESPONSE)	100	100-Hold
6-OFFER (RESPONSE)	110	110-Reject
6-OFFER (RESPONSE)	120	120-Withdrawn

- ER\_STS\_RSN\_TBL

Recruitment Area	Status Code	Status	Reason Description
1-JOB REQUISITION	010-Open	010	New authorization
1-JOB REQUISITION	010-Open	020	New Job/Position
1-JOB REQUISITION	010-Open	030	Temporary assignment
1-JOB REQUISITION	010-Open	040	Job/Position vacated
3-POSITION APPLIED FOR	073-Request Wrks Cncl Approval	010	Prehire
3-POSITION APPLIED FOR	080-Ready to Hire	010	Approved by wrks council
3-POSITION APPLIED FOR	110-Reject	010	Another applicant was hired
3-POSITION APPLIED FOR	110-Reject	020	Unable to contact

<b>Recruitment Area</b>	<b>Status Code</b>	<b>Status</b>	<b>Reason Description</b>
3-POSITION APPLIED FOR	110-Reject	030	Poor interview
3-POSITION APPLIED FOR	110-Reject	040	No skills match
3-POSITION APPLIED FOR	110-Reject	050	Underqualified
3-POSITION APPLIED FOR	110-Reject	060	No show for interview
3-POSITION APPLIED FOR	110-Reject	070	Misrepresentation
3-POSITION APPLIED FOR	110-Reject	080	Selected for other position
3-POSITION APPLIED FOR	110-Reject	090	Lacks required education
3-POSITION APPLIED FOR	110-Reject	100	Ineligible – Basic eligibility
3-POSITION APPLIED FOR	110-Reject	110	Ineligible – Employment Cond
3-POSITION APPLIED FOR	110-Reject	120	Ineligible - Min grade/salary
3-POSITION APPLIED FOR	110-Reject	130	Lacks other min qualifications
3-POSITION APPLIED FOR	110-Reject	140	Lacks required credentials
3-POSITION APPLIED FOR	110-Reject	150	Not selected
3-POSITION APPLIED FOR	110-Reject	160	Lacks required experience
3-POSITION APPLIED FOR	110-Reject	170	Requisition cancelled
3-POSITION APPLIED FOR	110-Reject	180	No opening
3-POSITION APPLIED FOR	110-Reject	190	Another job
3-POSITION APPLIED FOR	110-Reject	200	Offer rejected

Recruitment Area	Status Code	Status	Reason Description
3-POSITION APPLIED FOR	110-Reject	210	Rejected by works council
4-ROUTE	110-Reject	010	Insufficient experience
4-ROUTE	110-Reject	020	Insufficient skills
4-ROUTE	110-Reject	030	No opening
5-INTERVIEW	110-Reject	010	Base salary
5-INTERVIEW	110-Reject	020	Insufficient skills
6-OFFER	110-Reject	010	Base salary
6-OFFER	110-Reject	020	Benefits
6-OFFER	110-Reject	030	Other compensation
6-OFFER	110-Reject	040	Another job
6-OFFER	110-Reject	050	Personal

## Task I-1-2: Reviewing Profile Management Table Deletions

During the upgrade to Profile Management, there is data that will not be upgraded and thus will be not be in the new release. There is no action for you to take, as this data cannot fit into the new Profile Management model. Some of these tables contain related language data. The following tables are affected:

- CM\_EVALUAT\_LNG
- CM\_ELEM\_DTL\_LNG
- ACCOMPLISH\_LANG
- SCHL\_EDUC\_LANG

---

## Task I-2: Reviewing Records with Duplicate Key Structures

In PeopleSoft 7.5, the records listed below had keys that were defined as duplicate keys. In PeopleSoft 8.3, the duplicate key structure was removed from these records. Where a PeopleSoft 7.5 record has duplicate keys, the data conversion will eliminate all but the first row that it retrieves. If you want to preserve this data, you may run the following SQL, which will return all rows where the duplicate key condition is met. These SQLs need to be run on your current (old) database before the PeopleTools conversion takes place.

- INTERVIEW

```
SELECT *
FROM PS_INTERVIEW B2
WHERE EXISTS (SELECT 'X' FROM PS_INTERVIEW B
WHERE B.APPLID = B2.APPLID
```

```

AND B.APP_DT = B2.APP_DT
AND B.JOB_REQ_NBR = B2.JOB_REQ_NBR
AND B2.INTERVW_DT = B.INTERVW_DT
AND B2.FINAL_RECOMMENDATN = 'Y'
GROUP BY B2.APPLID, B2.APP_DT, B2.JOB_REQ_NBR
HAVING COUNT(*)>1)
SELECT *
FROM PS_INTERVIEW B
WHERE EXISTS (SELECT 'X' FROM PS_INTERVIEW B2
WHERE B.EMPLID = B2.EMPLID
AND B.JOB_REQUISITION# = B2.JOB_REQUISITION#
AND B.POSITION_NBR = B2.POSITION_NBR
AND B.SETID = B2.SETID
AND B.JOBCODE = B2.JOBCODE
AND B. INTERVW_DT = B2.INTERVW_DT
AND B2.FINAL_RECOMMENDATN = 'Y'
GROUP BY B2.EMPLID, B2.JOB_REQUISITION#, B2.POSITION_NBR, B2.SETID, B2.JOBCODE
HAVING COUNT (*)>1)

```

- **APPLICATN\_ROUTE**

```

SELECT *
FROM PS_APPLICATN_ROUTE B
WHERE EXISTS (SELECT 'X' FROM PS_APPLICATN_ROUTE A
WHERE A.APPLID = B.APPLID
AND A.APP_DT = B.APP_DT
AND A.JOB_REQ_NBR = B.JOB_REQ_NBR
AND A.ROUTE_DT = B.ROUTE_DT
AND A.ROUTE_TO_ID = B.ROUTE_TO_ID
GROUP BY A.APPLID, A.APP_DT, A.JOB_REQ_NBR, A.ROUTE_DT, A.ROUTE_TO_ID
HAVING COUNT(*) > 1)
SELECT*
FROM PS_APPLICATN_ROUTE B
WHERE EXISTS (SELECT 'X' FROM PS_APPLICATN_ROUTE B2
WHERE B.EMPLID = B2.EMPLID
AND B.JOB_REQUISITION# = B2.JOB_REQUISITION#
AND B.POSITION_NBR = B2.POSITION_NBR
AND B.SETID = B2.SETID
AND B.JOBCODE = B2.JOBCODE
AND B.ROUTE_DT = B2.ROUTE_DT
AND B.ROUTE_TO_ID = B2.ROUTE_TO_ID
GROUP BY B2.EMPLID, B2.JOB_REQUISITION#, B2.POSITION_NBR, B2.SETID, B2.JOBCODE,
B2.ROUTE_DT, B2.ROUTE_TO_ID

```

- **OFFER**

```

SELECT *
FROM PS_OFFER B
WHERE EXISTS (SELECT 'X' FROM PS_OFFER A

```

```

WHERE A.APPLID = B.APPLID
AND A.APP_DT = B.APP_DT
AND A.JOB_REQ_NBR = B.JOB_REQ_NBR
AND A.OFFER_DT = B.OFFER_DT
    GROUP BY A.APPLID, A.APP_DT, A.JOB_REQ_NBR, A.OFFER_DT
    HAVING COUNT (*) > 1)
SELECT *
FROM PS_OFFER B
WHERE EXISTS (SELECT 'X' FROM PS_OFFER B2
WHERE B.EMPLID = B2.EMPLID
AND B.JOB_REQUISITION# = B2.JOB_REQUISITION#
AND B.POSITION_NBR = B2.POSITION_NBR
AND B.SETID = B2.SETID
AND B.JOBCODE = B2.JOBCODE

```

## Task I-3: Understanding Applicant Data Normalization

In the new Recruit Workforce release, Applicant Records were normalized. In PeopleSoft 8.1, the high level record in the Application realm was APPLICANT\_DATA, and the high level keys were APPLID and APP\_DT. In PeopleSoft 8.3 an APPLICANT record is introduced to abstract the concept of an applicant (APPLICANT) from the concept of the application (APPLICANT\_DATA). The new APPLICANT record has one key: APPLID. An effective-dated Applicant record, APPL\_DATA\_EFFDT, was also introduced to maintain date-sensitive Applicant information, such as address information.

Several of the lower-level application records had to be modified to complete the normalization process. Specifically, the records below are determined to be children of the APPLICANT entity and not the APPLICATION entity. Because of this, APP\_DT was removed from the key structure of these records. The logic that the data conversion applied was the following: Delete rows from these records where the APP\_DT is not the maximum date less than or equal to the system date (similar to Current Effective Dating logic).

For example, here is the SQL that would return the rows that are going to be deleted in the APPLICNT\_PHONE record:

```

SELECT *
FROM PS_APPLICNT_PHONE A
WHERE A.APP_DT <> (
    SELECT MAX (B.APP_DT)
    FROM PS_APPLICNT_PHONE B
    WHERE B.APPLID = A.APPLID
    AND B.PHONE_TYPE = A.PHONE_TYPE
    AND B.APP_DT <= %CurrentDateIn)

```

Here is a complete list of the records for which this logic applies. If you would like to store this data, please run a SQL, similar to the SQL above and print the data for your records.

8.1 Record Name
APPLICNT_PHONE

8.1 Record Name
APP_ACCOM_OPTN
APP_ACCOM_RQST
APP_ACM_OPT_LNG
APP_BANKACCT
APP_CITIZENSHIP
APP_CITIZN_PSPT
APP_CIT_PSP_LNG
APP_DISABILITY
APP_DISABL_CHE
APP_DISABL_FRA
APP_DISABL_GER
APP_DIVERSITY
APP_EMAIL_ADDRS
APP_NATNLTY_GER
APP_PL_ORIG_CHE
APP_VSA_PMT_DTA
APP_VSA_PMT_LNG
APP_VSA_PMT_SPT

## Task I-4: Reviewing the National ID Expiration Record

In the new PeopleSoft release, the National ID expiration record, PERS\_NID\_EXP used to store Canadian expiration dates, now only contains data for National ID rows that have an expiration date. If an expiration date does not exist, a row will not be stored in this record as it was in previous releases. If you have customized views or reports that rely on a one-to-one relationship between a Canadian National ID row in PERS\_NID and PERS\_NID\_EXP this change will affect you.

## Task I-5: Reviewing Schedule Defaults

This section discusses:

- Reviewing Shift and Workdays Upgrade
- Determining the Effective Date of Schedule Definition
- Reviewing Schedule Override Tables

### Task I-5-1: Reviewing Shift and Workdays Upgrade

Review the list below to see how the upgrade modified Shifts and Workdays.

- The effective date of the Shift IDs and Workday IDs are set to the earliest Effective Date from the Schedule Definition, in which they are used, only if the Schedule Definition has an earlier effective date than that of the Shift or Workday.
- The setID is set to the setID of the Schedule Definition in which they are used.
- If the Shifts and Workdays are used in multiple Schedule Definitions with different setIDs, a copy of the Shift and Workday are created for each of the setIDs.

### Task I-5-2: Determining the Effective Date of Schedule Definition

The upgrade process determines the Effective Date of the Schedule Definitions. It begins by determining the minimum effective date of where the Schedule ID is used. This is called the “First Date Used.” The locations to check for the use of the Schedule ID are:

- Assign Work Schedule (SCH\_ASSIGN) as the Schedule ID or the Alternate Schedule ID.
- Global Payroll Paygroup (GP\_PYGRP\_DTL) as the Schedule ID or the Alternate Schedule ID.
- Workgroup (TL\_WRKGRP\_TBL) as the Schedule ID.

Follow the steps below to determine the effective date of the Schedule Definition:

1. Check the last date that was built for the Schedule ID from the Schedule Calendar (SCH\_CLND\_TBL) by checking the View Schedule Calendar page in the source database.
2. Determine the number of days for that Schedule Definition.  
This process is done after converting the rest of the Schedule Definition so the Num Days in Schedule is already determined.
3. Add one day then start subtracting the “Number of Days” for that Schedule Definition, to get the Start Day of the last schedule pattern of the Schedule Calendar.
4. Continue subtracting by that “Number of Days” for that schedule, until the systems derives the date on or before the “First Date Used” determined earlier.

For example, Schedule ID KUSCHDFP1 has a First date used=Schedule ID KUSCHDFP1.

- 1/1/1980 is the earliest effective date used in a Workgroup KUWRKGRP1.
- 9/25/1982 is the earliest effective date used for EmplID KU0020 in SCH\_ASSIGN as Schedule ID



The last date of the schedule calendar for KUSCHDFP1 Schedule Definition = 12/30/2006 from the PS\_SCH\_CLND\_TBL. Add one day, then subtract the number of days in schedule (Duration of schedule = seven days) from 12/31/2006 = 12/24/2006. Continue subtracting the “Num Days in Schedule” until you get a date to begin the schedule pattern, which is on or before the “First Day Used” = 12/30/1979 = Sunday.

## Task I-5-3: Reviewing Schedule Override Tables

The following list explains how the Schedule Override tables are upgraded in the new PeopleSoft release:

- For the SCH\_MNG\_SCH\_TBL which was the storage table for the Override Schedule Workday, Shift ID has been added to this table. The Shift ID will be populated from the old Workday details.
- If a Workday had consisted of more than one shift, then a new row would be inserted into SCH\_MNG\_SCH\_TBL for each of the Shifts, whether a Primary or Alternate Workday override.
- The population of Sched Hrs, Sch Config Totals, and Start and End Times are all populated from the Shift Tables.
- The SETID is set to the SETID of the Workday.
- If there is an Alternate Workday override from PeopleSoft 8 SP1, there will be another row in the SCH\_MNG\_SCH\_TBL for the Alternate override.
- The SCH\_MNG\_SCH\_TBL is now populated for any third party overrides that exist in SCH\_ADHOC\_DTL. (In previous PeopleSoft releases any Scheduling third party interface populated only SCH\_ADHOC\_DTL).
- The SCH\_ADHOC\_DTL table is now also populated for the Alternate as well as the Primary Schedule Overrides, using details from the Shift table. (In previous PeopleSoft releases, this was only populated for Primary Workday overrides.)



## APPENDIX J

# Sizing Tables for the Upgrade

This appendix discusses:

- Sizing Tables

---

### Task J-1: Sizing Tables

Many of the new tables and indexes will be populated during the upgrade. If they are not sized appropriately for your database, the conversion programs will stop with errors. The following lists these new tables with information about how to size them relative to the size of your existing data. This information is meant to give you a rough estimate on an initial size so you can execute the conversion programs without errors.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PY	PS_ALTOT_COMP_TMP	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>
PY	PS_ALTOT_JOBAUSTMP	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PY	PS_ALTOT_JOBINDTMP	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>
PY	PS_ALTOT_JOBKEYS	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>
PY	PS_ALTOT_JOBUSFTMP	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion and you have the Federal HRMS product.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>
PY	PS_ALT_OT_STATES	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion. This is a setup table and would be expected to be relatively low volume.</p> <p>One row for each state allowing alternative overtime per setID, as specified on the Upgrade Defaults page.</p>

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
HR	PS_APPL_DATA_EFFDT	One row per applicant per application.
HR	PS_APPLICANT	One row per applicant.
TL	PS_AUDIT_SCH_TBL	Audit Scheduling table.
TL	PS_AUDIT_TL_PAY_TM	Audit payable time.
TL	PS_AUDIT_TLRPTTIME	All the audit data will be moved to this new table.
BN	PS_BEN_CONTACT_HDR PS_BEN_CONTACT_TBL	This is a setup table and would be expected to be relatively low volume.  Approximately the same number of records as in PS_COBRA_ADMINISTR.
BN	PS_BN_FORM_DEF	Approximately 1-10 rows for each parent PS_BN_FORMULA row.
BN	PS_BN_FORMULA	Approximately the same number of rows as PS_LIFE_ADD_TBL and PS_DISBLTY_PLN_TBL combined.
BN	PS_BN_RATE_DATA	Approximately the same number of rows as PS_FLAT_RATE_TBL, PS_SALARY_RATE_TBL, PS_SVC_COVG_TBL, and 2 x PS_AGE_COVG_TBL (for Smoker vs Non-smoker) combined.
BN	PS_BN_RATE_TBL	Will have the same number of rows as PS_AGE_RATE_TBL, PS_FLAT_RATE_TBL, PS_SALARY_RATE_TBL, and PS_SVC_RATE_TBL combined.
BN	PS_BN_RATE_TBL_LNG	For each row in parent PS_BN_RATE_TBL, will have one row for each non-base language supported by the client.
BN	PS_COVRG_RULE_TBL	This is a small, fairly static setup table and will have approximately one to three times the number of rows as in PS_COVRG_CD_TBL.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
BN	PS_DEP_BEN PS_DEP_BEN_ADDR PS_DEP_BEN_EFF PS_DEP_BEN_NAME	All four new tables will each have the same number of rows as exist in PS_DEPENDENT_BENEF. After upgrade, PS_DEPENDENT_BENEF will become a view on these tables, rather than be a physical table.
BN	PS_ENROLL_EXCEPTN	Rows with withdrawal dates from PS_SAVINGS_MGT_EE.
EP	PS_EP_APPR_B_ITEM	One row for each item within each appraisal if using the baseline documentation.
EP	PS_EP_APPR_B_ROLE	One row for each role within each appraisal if using the baseline documentation.
EP	PS_EP_APPR_B_SEC	One row for each section within each appraisal if using the baseline documentation.
EP	PS_EP_APPR_B_SUB	One row for each sub-item within each appraisal if using the baseline documentation.
EP	PS_EP_APPR_NOMINEE	One row for each reviewer nominated to participate in a review, per appraisal.
EP	PS_EP_APPR_NOM_LIST	One row for each person selected to participate in a review, per appraisal.
EP	PS_EP_APPR_PARTIC	One row for each role type participating in a review, per appraisal.
EP	PS_EP_ROLE_LANG	Related Language for role setup table. Size should be small.
EP	PS_EP_ROLE_TBL	Setup table. Use to set up role types. Size should be small.
EP	PS_EP_TMPL_PARTIC	Used to create/update document template participation roles. One row per review type, template ID and role.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
FA	PS_FSA_SUMMARY	One row per employee, per year, for each Flexible Spending Account plan (for example, Health Care or Dependent Care) for which the employee submitted claims.
PY	PS_GARN_EXEMPT_TBL	This is not a new table. However, upgrade processes will insert many additional rows during data conversion. It will contain a summary of rows for U.S. and Canadian garnishment rules maintained by PeopleSoft (defined as system data in PeopleSoft 9), plus all custom rows for U.S. and/or Canadian garnishment rules that have been added in the prior release.
PY	PS_GARN_OPERANDS	This is not a new table. However, upgrade processes will insert many additional rows during data conversion. It will contain a summary of rows for U.S. and Canadian garnishment rules maintained by PeopleSoft (defined as system data in PeopleSoft 9), plus all custom rows for Canadian garnishment rules defined in the Copy of Production, plus 6 rows on average for each custom U.S. garnishment rule from the Copy of Production database.
GP	PS_GPGB_EE_IN	Will have the same number of rows as the existing PS_GPGB_EE_STDtls table.
GP	PS_GPGB_EE_TAX	Will have the same number of rows as the existing PS_GPGB_EE_STDtls record.
PM	PS_JPM_CAT_GMEMB	The same number of rows as PS_COMPETENCY_TYPS.
PM	PS_JPM_CAT_GROUPS	The same number of rows as PS_CM_TYPE_TBL.
PM	PS_JPM_CAT_GRP_LNG	The same number of rows as PS_CM_TYPE_LANG.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PM	PS_JPM_JP_SYND_SRC	The number of rows will be the same as the number of cluster profiles syndicated to role profiles.
HR	PS_PERSON	The same number of records as in PS_PERSONAL_DATA.
PY	PS_PYGL_ACTIVDEDDT	Approximately the same number of rows as the Deductions Activity Group.
PY	PS_PYGL_ACTIVDEDMP	Approximately one row per Company and Deductions Activity Group for each Deductions Code within that group that uses the same Expense Account.
PY	PS_PYGL_ACTIVERNDT	Approximately the same number of rows as the Earnings and Activity groups.
PY	PS_PYGL_ACTIVERNMP	Approximately one row per Company and Earnings Activity Group, for each Earnings Code within that group that uses the same Expense Account.



Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PY	PS_PYGL_ACTIVITYGP	<p>Summary of Earnings, Deductions, and Taxes Activity Groups.</p> <p>Earnings Activity Groups: Approximately one row per Company for each Activity defined in PS_CHART_KEYS_HR that is used as a distinct Expense Account in PS_EARNINGS_TBL.</p> <p>Deductions Activity Groups: Approximately one row per Company for each Activity defined in PS_CHART_KEYS_HR that is used as a distinct Expense Account in PS_DEDUCTION_CLASS</p> <p>Taxes Activity Groups: Approximately one row per USA Company for each Activity defined in PS_CHART_KEYS_HR that is used as a distinct Expense Account in PS_COMPANY_TBL_GL and/or one row per Canadian Company, Wage Loss Plan, and each Activity defined in PS_CHART_KEYS_HR that is used as an Expense Account in PS_WGLS_EXP_GL_ACC.</p>
PY	PS_PYGL_ACTIVTAXDT	Approximately the same number of rows as the Taxes Activity groups.
PY	PS_PYGL_ACTIVTAXMP	Approximately one row per Company and Taxes Activity Group for each Tax Class within that group that uses the same Expense Account.
PY	PS_PYGL_CF_CPNYCTX	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit and has chartkeys defined for the CAN Payroll Taxes Activity Group.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PY	PS_PYGL_CF_CPNYDED	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit and has chartkeys defined for the Deductions Activity Group.
PY	PS_PYGL_CF_CPNYERN	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit and has chartkeys defined for Earnings Activity groups.
PY	PS_PYGL_CF_CPNYLEV	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit.
PY	PS_PYGL_CF_CPNYTAX	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit and has chartkeys defined for the US Payroll Taxes Activity Group.
PY	PS_PYGL_CF_MAPPING	Approximately one row per Company for each HR Business Unit that is mapped to a GL Business Unit.
PY	PS_PY_TAX_GL_CLASS	This is a setup table and would be expected to be relatively low volume.  One row for each Payroll Activity and Account Type for U.S.A. and Canada.
BN	PS_SAVINGS_MGT_EE	Essentially the same size as the existing PS_SAVINGS_403B_EE table (which will now be static, containing only historical data).
HR	PS_SP_MER_INCR_EE	For each employee, one row for each “current” Employee Review entry for each Review Type that is associated with a salary/merit increase.

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
HR	PS_SP_MER_INCR_RNG	This is a setup table and would be expected to be low volume.  One row for each Merit Group Table entry associated with a merit scale represented in the new table PS_SP_MER_INCR_TBL.
HR	PS_SP_MER_INCR_TBL	This is a setup table and would be expected to be low volume.  One row for each Review Scale Table entry of the type <i>Summation</i> .
TL	PS_TL_RPTD_TIME	All the data from PS_TL_RPTD_ELPTIME and PS_TL_RPTD_PCHTIME will be moved to this new table.
TL	PS_TL_SCH_EXT_DTL	Scheduling external detail table.
TL	PS_TL_TR_COMMENTS	If you used comments in time reporting, then this table will have comments rows for both Punch and Elapsed data tables.
HP	PS_UPG_ACCT_CD_TAO	Approximately the number of distinct ACCT_CD values in PS_UPG_ACCT_CD_TBL (renamed PS_ACCT_CD_TBL).
PY	PS_UPG_ACTBU_TAO	Approximately the number of distinct ACCT_CD values in PS_UPG_ACCT_CD_TBL (renamed PS_ACCT_CD_TBL) multiplied by the number of GL Business Units you use.
PY	PS_UPG_ACTCD_L_TAO	Approximately the number of distinct ACCT_CD values in PS_UPG_ACCT_CD_TBL (renamed PS_ACCT_CD_TBL).

Product Code	Table Populated During Upgrade	Estimate of Number of Rows of Data Inserted
PY	PS_UPG_HRSNRDTSTMP	<p>This table will be populated if you make functional decisions to do Alternative Overtime conversion.</p> <p>Approximately one row per employee/employee record for those employees whose Job records currently have FLSA status equal to Nonexempt, HR status not equal to Terminated, and who work in a state specified on the Alt Overtime upgrade default page.</p>
HR	PS_UPG_PERSDTF_TAO	Approximately the same number of records as in PS_PERS_DATA_EFFDT.
HP	PS_UPG_VALCMBO_TAO	Approximately the same number of records as in PS_UPG_ACCT_CD_TBL (renamed PS_ACCT_CD_TBL).
HP	PS_VALID_COMBO_LNG	Approximately the same number of records as in PS_UPG_ACCT_CD_LNG (renamed PS_ACCT_CD_TBL_LNG).
HP	PS_VALID_COMBO_TBL	Approximately the same number of records as in PS_UPG_ACCT_CD_TBL (renamed PS_ACCT_CD_TBL).

## APPENDIX K

# Understanding Core Record Modifications

## Understanding Core Record Modifications

The following table summarizes the changes in the new PeopleSoft HRMS Core Record.

Table	Change	Field	Label	Upgrade Notes
PERS_DATA_ EFFDT	Delete	QDRO_IND_YN	QDRO Indicator Code	Value of the field is derivable from the data in PA_QDRODAT so there is no need to keep the information in PERS_DATA_ EFFDT.  Logic:  If there is a PA_QDRODAT record that has the person entered in the QDRO_ID field, then the QDRO indicator would be A else if there is a PA_QDRODAT record that has the person entered in the EMPLID field, then the QDRO indicator would be Q, else QDRO indicator would be blank.
DISABILITY	Add	DISABILITY_GR_JPN	Disability Grade	New Field
DISABILITY	Add	DISABILITY_CD_JPN	Disability Code	New Field
DISABILITY	Add	WKRCOMP_DISGP_JPN	Worker Comp Disability	New Field
DISABILITY	Add	WKRCOMP_ISGRD_JPN	WC Injury and Sickness	New Field

Table	Change	Field	Label	Upgrade Notes
PER_ORG_ASG_BRA	Add	INS_DAYS_BRA	INSS Days	New Field
PER_ORG_ASG_BRA	Add	INS_MONTHS_BRA	INSS Months	New Field
PER_ORG_ASG_BRA	Add	INS_YEARS_BRA	INSS Years	New Field
JOB	Add	APPT_TYPE	Appointment Type – used to distinguish between a French Public Sector row, an Additional Appointment for JPNrow and a normal core row.	New Field
JOB	Add	MAIN_APPT_NUM_JPN	Main Appt Number of an Additional Appointment Row for JPN	New Field
JOB	Delete	RATING_SCALE	Rating Scale	Information now kept in Performance Management Records by Group – see REVIEW_DISTRIB.
JOB	Delete	REVIEW_DT	Review Date	Information now kept in Performance Management Record in a difference manner– EE_APPR.

Table	Change	Field	Label	Upgrade Notes
JOB	Delete	REVIEW_RATING	Review Rating	Information now kept in Performance Management Record in a different manner – EE_APPR.
JOB	Delete	SALARY_MATRIX_CD	Salary Matrix Code	Information now kept in Performance Management Records by Group – see REVIEW_DISTTRIB.
JOB	Delete	WRKS_CNCL_ID_LCL	Works Council ID	Field renamed WC_COUNCIL_ID  Data is now kept at the Company Location Level.
JOB	Delete	WRKS_CNCL_MEM_GER	Works Council Member	Data is now kept at the Company Location Level.
JOB	Change	FTE	FTE	Decimal Pos increased to 6
JOB	Change	PAID_FTE	Paid FTE	Decimal Pos increased to 6
JOB_JR	Add	START_DATE_JPN	JPN Start Date	New Field
JOB_JR	Add	EXP_END_DATE_JPN	JPN Expected End Date	New Field
JOB_JR	Add	END_DATE_JPN	JPN Actual End Date	New Field
JOB_JR	Add	WORKDAYS_NLD	NLD Number of Working days per week	New Field
JOB_JR	Add	CONT_SAL_TYPE_BRA	BRA Contract Salary Type	New Field
JOB_JR	Add	CBO_CD_BRA	BRA Occupation Code	New Field
JOB_JR	Add	SEFIP_CATEGORY_BRA	BRA SEFIP Category	New Field

Table	Change	Field	Label	Upgrade Notes
JOB_JR	Add	UNION_CD_ALT_BRA	BRA Alternate Union	New Field
JOB_JR	Add	SUSBTIT_REASON_ESP	ESP Substitution Reason	New Field
JOB_JR	Add	SUSBTIT_EMPLID_ESP	ESP Substitution EMPLID	New Field
JOB_JR	Add	PART_SUBROGTN_ESP	ESP Partial Subrogation	New Field
JOB_JR	Add	LABOR_RLTNSHIP_ESP	ESP Special Labor Relations	New Field
COMPANY_TBL	Delete	GL_FWT	US GL Acct# - FWT	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FICA_EE_OASDI	US GL Acct# - FICA OASDI EE	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FICA_EE_MED	US GL Acct# - FICA Medicare EE	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FICA_ER_OASDI	US GL Acct# - FICA OASDI ER	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FICA_ER_MED	US GL Acct# - FICA Medicare ER	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FUT	US GL Acct# - FUT	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_EIC	US GL Acct# - EIC	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_NET_PAY	US GL Acct# - Net Pay	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_DIR_DEPOSITS	US GL Acct# - Direct Deposits	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FICA_EXP_OASDI	US GL Acct# FICA OASDI Exp	Moved to COMPANY_TBL_GL



Table	Change	Field	Label	Upgrade Notes
COMPANY_TBL	Delete	GL_FICA_EXP_MED	US GL Acct# - FICA Medicare Exp	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_FUT_EXPENSE	US GL Acct# - FUT Expense	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_SUT_EXPENSE	US GL Acct# - SUT Expense	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_SDI_EXPENSE	US GL Acct# - SDI Expense	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_VDI_EXPENSE	US GL Acct# - VDI Expense	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	GL_LCL_EXP	US GL Acct# - LCL Expense	Moved to COMPANY_TBL_GL
COMPANY_TBL	Delete	WRKS_CNCL_REQ_GER	GER Works Council Administration Required	
JOBCODE_TBL	Add	JOB_PROFILE_ID	Job Profile ID	New Field
LOCATION_TBL	Delete	WRKS_CNCL_ID_LCL	Works Council ID	Data is now kept at the Company Location Level.
ACTN_REASON_TBL	Delete	WRKS_CNCL_APPR_GER	Works Council Approval Required	No upgrade – data delivered as system data.
ACTN_REASON_TBL	Delete	DECISIONGRP_GER	GER Decision Group	Field renamed WC_DECISIONGRP_ID  No upgrade – data delivered as system data.
ACTN_REASON_TBL	Add	AFI_EE_STATUS_ESP	ESP Employee Status	New Field
ACTN_REASON_TBL	Add	AFI_SS_STATUS_ESP	ESP Social Security Action	New Field

Table	Change	Field	Label	Upgrade Notes
ESTAB_TBL	Add	FONGECIF_CD_FRA	FRA FONGECIF Code	New Field
ESTAB_TBL	Add	ESTAB_CONTACT_ID	FRA Contact for Establishment	New Field
INSTALLATION	Add	IND	HR Country India	New Field
INSTALLATION	Add	BRA	HR Country Brazil	New Field
INSTALLATION	Add	INSTALLED_GP_IND	GP Country India Product	New Field
INSTALLATION	Add	PSEPERF_MANAGEMENT	PS ePerformance Management Product	New Field
INSTALLATION	Add	TL_MOBILE	PS Mobile Time Mgt Product	New Field
INSTALLATION	Add	AA_ENABLED_JPN	Additional Appointment for JPN enabled Flag	New Field

## APPENDIX L

# Understanding Dynamic ChartFields Changes

This appendix discusses:

- Understanding Dynamic ChartFields Upgrade
- Understanding Changes in the Account Code Table
- Understanding Changes in GL Interface Setup
- Upgrading Time and Labor ChartFields

---

## Understanding Dynamic ChartFields Upgrade

In PeopleSoft 9.0 there have been changes in the way Payroll for North America uses Standard ChartField configuration. This new functionality will vary according to the financials system being used, but no matter what financials system you use, the upgrade will impact your existing data.

---

**Note.** Changes in General Ledger Interface Setup impact you only if you use non-commitment accounting, while changes in Account Code Table impact you no matter what type of accounting you use.

---

---

## Task L-1: Understanding Changes in the Account Code Table

This section discusses:

- Understanding Combination Codes
- Understanding FDM\_HASH

### Task L-1-1: Understanding Combination Codes

Combination Code represents a particular set of ChartField values. The ChartField values used to be stored in ACCT\_CD\_TBL.

In PeopleSoft 9.0 the structure of ACCT\_CD\_TBL is changing as well as the content. The new ACCT\_CD\_TBL will store only combination codes used in Payroll for North America for either commitment accounting or noncommitment accounting.

In PeopleSoft 9.0 there is a new table VALID\_COMBO\_TBL that will store combination codes as they are defined in the Financials database.

Both tables will be populated during the upgrade process with the data from your ACCT\_CD\_TBL as follows:

- Current ACCT\_CD\_TBL will be physically renamed into UPG\_ACCT\_CD\_TBL.

- Current ACCT\_CD\_TBL\_LNG will be physically renamed into UPG\_ACCT\_CD\_LNG.
- To avoid confusion between DEPTID used in the Financials database and DEPTID used in the HCM database, the DEPTID in CHARTFLD\_KY\_SBR will be renamed into DEPTID\_CF.

New tables will have key structure as follows:

- ACCT\_CD\_TBL:
  - FDM\_HASH – a new value built for efficient access of the data. It is not going to be visible on any of the pages.
- VALID\_COMBO\_TBL:
  - SETID – value converted from UPG\_ACCT\_CD\_TBL.SETID.
  - PROCESS\_GROUP – blank value.
  - FDM\_HASH – a new value built for efficient access of the data. It is not going to be visible on any of the pages.
  - EFFDT – value converted from UPG\_ACCT\_CD\_TBL.EFFDT.

ACCT\_CD, the old key in ACCT\_CD\_TBL, will still be saved in both tables; however, since it does not represent the same values in both tables, it will be saved as follows:

- ACCT\_CD in ACCT\_CD\_TBL
- FDM\_COMBO\_CD in VALID\_COMBO\_TBL

You will be able to use these fields to reconcile your conversion and verify that all data are converted as expected.

Both tables will be populated during the upgrade, but that does not mean you have to use them both in your new release. Using ACCT\_CD\_TBL is mandatory, but using VALID\_COMBO\_TBL is optional and you can clear the table if you are not planning to use it. However, the timing of deleting these entries is crucial and might impact your setup of Expenses ChartField Mapping if you are planning to use that functionality. Please keep in mind that ACCT\_CD\_TBL in a new release might have less entries than you had in the old release because it will only have entries you actually used. All entries from VALID\_COMBO\_TBL that you are planning to use in a new release will be saved in ACCT\_CD\_TBL during the Expenses ChartField Mapping, if they have not been used already. If you clear VALID\_COMBO\_TBL before you complete the setup, you might not have all expected entries available for a setup.

---

**Note.** Recommendation: Ensure all combination codes you need for setting up Expenses ChartField Mapping in a new release are entered in your old release before you make a copy of production database.

---

## See Also

*PeopleSoft Application Fundamentals for HRMS PeopleBook, for your new release.*

*PeopleSoft Payroll for North America PeopleBook, for your new release.*

*PeopleSoft Human Resources PeopleBook: Manage Commitment Accounting, for your new release.*

## Task L-1-2: Understanding FDM\_HASH

FDM\_HASH is a unique value representing a combination of all 16 ChartFields. It consists of two parts: characters 1-28 are hash value built based on ChartField combination, characters 29-31 are sequence number added to hash value to ensure uniqueness of the key.

FDM\_HASH is dependent on the format of each ChartField; therefore, the format of each ChartField in UPG\_ACCT\_CD\_TBL is updated to uppercase to ensure that hash value will be built correctly.

---

**Note.** Each record in UPG\_ACCT\_CD\_TBL that has SETID and at least one ChartField populated will be inserted into VALID\_COMBO\_TBL.

---

It is possible that the same ChartField combination is represented with two different account codes in the UPG\_ACCT\_CD\_TBL:

- If the second (and any following) identical ChartField combination is within the same SETID, the sequence number of the FDM\_HASH will be incremented by 1. If any of these identical ChartField combinations has 'Active' Effective Status, all of them will have Effective Status updated to 'Active'.
- If the second (and any following) identical ChartField combinations is within two different SETIDs, the sequence number of the FDM\_HASH will not be incremented. This will result in two or more records with identical FDM\_HASH, but it is valid because FDM\_HASH is just one of the keys and the inserted records will not be duplicates.

It is possible that the algorithm used to create FDM\_HASH returns the same value for two different ChartField combinations:

- If the second ChartField combination is within the same SETID, the sequence number of the FDM\_HASH will be incremented by 1.
- If the second ChartField combination is within two different SETIDs, the sequence number of the FDM\_HASH will not be incremented.

---

**Note.** Each record in UPG\_ACCT\_CD\_TBL that has been used for distribution anywhere in PeopleSoft HCM and has at least one ChartField populated will be inserted into a new ACCT\_CD\_TBL.

---

FDM\_HASH on ACCT\_CD\_TBL should be in sync with FDM\_HASH on VALID\_COMBO\_TBL in the majority of the cases, but there are some exceptions to this:

- It is possible that the ChartField combination doesn't exist on VALID\_COMBO\_TBL, but because it was used for distribution it has to be converted into a new structure of ACCT\_CD\_TBL. ChartField combinations that have blank SETID are skipped during VALID\_COMBO\_TBL conversion.
- It is possible that FDM\_HASH is duplicated on the new ACCT\_CD\_TBL even though it is not duplicated on VALID\_COMBO\_TBL. VALID\_COMBO\_TBL has unique FDM\_HASH within SETID. If there are two or more records with different account codes, but the same ChartField combination within different SETIDs, they will not be duplicates in VALID\_COMBO\_TBL, but will be duplicates in ACCT\_CD\_TBL. To avoid this problem with duplicates and keep the distribution history as is, the sequence number of the FDM\_HASH will be increased by 1.

---

## Task L-2: Understanding Changes in GL Interface Setup

This section discusses:

- Understanding GL Interface Setup
- Understanding Liability Accounts
- Understanding GL Activity Grouping

- Understanding Expenses ChartField Mapping

## Understanding GL Interface Setup

In PeopleSoft 9.0, there have been changes to the General Ledger Interface setup procedure for organizations that use non-commitment accounting. There are some new interface tables introduced that provide more flexible integration between Payroll for North America and General Ledger.

Chart Dist HR and Chart Keys HR will not be used in PeopleSoft 9.0. Instead, new tables are defined to group payroll activities and map defined groupings to specific ChartFields. Liability offset accounts are defined on the payroll source table directly.

Chart Dist HR and Chart Keys HR can be deleted at the end of the conversion.

The fields that represent Expenses Account Type in the COMPANY\_TBL\_GL, WGLS\_EXP\_GL\_ACC, EARNINGS\_TBL, and DEDUCTION\_CLASS tables will not be physically deleted, but they are not accessible from any of the pages.

---

**Important!** Once you create your Copy of Production database during the initial pass you should not make any changes in your production database for Company, Company State Taxes, Company Local Taxes, Wage Loss Plan Table, and Deduction Table until you complete the upgrade. If you make changes to any of the above tables you will not be able to use the import script during the Move to Production pass to import the data that you have set up in the initial pass. You will either have to synchronize the data between production and the copy of the production in the initial pass before you set up data and complete the export, or manually set up data in your Copy of Production during the Move to Production pass.

---

## Task L-2-1: Understanding Liability Accounts

The General Ledger Interface process enforces balancing of GL transactions by creating offset credit transactions using Liability accounts. In PeopleSoft 9.0, these liability accounts are defined directly on payroll source tables.

If you are using non-commitment accounting, you must do following:

- Run and review Liability ChartField Usage audit report on your copy of the production during initial pass. Work in conjunction with general ledger staff to make sure the reviewed liability accounts are correctly assigned.

See “Prepare Your Database,” Running Application Audits, Running Dynamic ChartFields Audits.

- Manually update tables with liability accounts, if any updates are needed.

See “Complete Database Changes,” Setting Up General Ledger Interface, Setting Up the Liability Accounts.

- Export tables updated in your copy of the production during the initial pass of the upgrade.

See “Apply Application Changes,” Completing Application Processes, Exporting GL Interface Setup Tables.

- Import tables into your new copy of the production during the Move to Production (MTP) phase.

See “Apply Application Changes,” Completing Application Processes, Importing GL Interface Setup Tables.

Sections of audit reports listed under table name will help to decide how you want to set up data in each table:

- Company — COMPANY\_TBL\_GL

Review the liability accounts listed in the Company Federal Tax and Net Pay Liability ChartFields section of the Liability ChartField Usage report and set up the data by entering liability accounts for each Company if the update is needed.

The fields you might need to set up are as follows:

Field Name	Description
GL_NET_PAY	Net Pay
GL_DIR_DEPOSITS	Direct Deposit
GL_FWT	Federal Withholding
GL_EIC	Earned Income Credit
GL_FICA_EE_OASDI	OASDI – Employee
GL_FICA_EE_MED	FICA Medicare – Employee
GL_FICA_ER_OASDI	OASDI – Employer
GL_FICA_ER_MED	FICA Medicare – Employer
GL_FUT	Federal Unemployment

**Note.** NetPay and Direct Deposit liability accounts are for both US Payroll and Canadian Payroll. Federal Tax liability accounts are for US Payroll only.

- Company State Tax Table — CO\_STATETAX\_TBL

Review the liability accounts listed in the State Tax Liability ChartFields section of the Liability ChartField Usage report and set up the data in Company State Tax Table by entering liability accounts for each Company/State combination if the updates are needed.

The fields you might need to set up are as follows:

Field Name	Description
GL_SWT	State Withholding
GL_SUT	State Unemployment – Employee
GL_SUT_ER	State Unemployment – Employer
GL_SDI	State Disability – Employee
GL_SDI_ER	State Disability – Employer
GL_VDI_EE	Voluntary Deduction – Employee
GL_VDI_ER	Voluntary Deduction – Employer
GL_ST_EIC	State Earned Income Credit

---

**Note.** State Tax liability accounts are for US Payroll only.

---

- Company Local Tax Table — CO\_LOCALTAX\_TBL

Review the liability accounts listed in the Local Tax Liability ChartFields section of the Liability ChartField Usage report and set up the data in Company Local Tax Table by entering liability accounts for each Company/State/Locality combination if the updates are needed.

The fields you might need to set up are as follows:

Field Name	Description
GL_LWT	Local Withholding – Employee
GL_LWT_ER	Local Withholding – Employer

---

**Note.** Local Tax liability accounts are for US Payroll only.

---

- Wage Loss Plan Table — WGLS\_LIA\_GL\_ACC

Review the liability accounts listed in the Canadian Tax Liability ChartFields section of the Liability ChartField Usage report and set up the data in Wage Loss Table by entering liability accounts for each Wage Loss Plan/Tax Class combination if the updates are needed.

The fields you might need to set up are as follows:

Field Name	Tax Class	Description
GL_NBR	CIT	Canadian Income Tax
GL_NBR	CPP	Canadian Pension Plan – Employee
GL_NBR	CPR	Canadian Pension Plan – Employer
GL_NBR	EIE	Employment Insurance - Employee
GL_NBR	EIR	Employment Insurance - Employer
GL_NBR	HTX	Health Tax
GL_NBR	PYT	Payroll Tax
GL_NBR	QBT	Quebec Bonus Tax
GL_NBR	QIT	Quebec Income Tax
GL_NBR	QPP	Quebec Pension Plan – Employee
GL_NBR	QPR	Quebec Pension Plan – Employer



Field Name	Tax Class	Description
GL_NBR	RV2	Quebec Income Tax – RL2
GL_NBR	T4A	Canadian Income Tax – T4A

---

**Note.** Canadian Tax liability accounts are for Canadian Payroll only.

---

- Deduction Table — DEDUCTION\_CLASS

Review the liability accounts listed in the Deduction Liability ChartFields section of the Liability ChartField Usage report and set up the data in Deduction Table for each Wage Deduction Code/Deduction Class combination, if updates are needed.

The fields you might need to set up are:

Field Name	Description
GL_LIABILITY	Deduction Liability

---

**Note.** Deduction liability accounts are for both US Payroll and Canadian Payroll.

---

## Task L-2-2: Understanding GL Activity Grouping

In PeopleSoft 9.0, organizations that use non-commitment accounting can track expenses by using payroll activities groups. In prior releases, GL accounts were assigned to individual earning, deduction, and tax activities. In PeopleSoft 9.0, organizations can group payroll activities that share similar characteristics and that are assigned to one ChartField combination.

If you are using non-commitment accounting you must do the following:

- Run and review Chartkey Usage by Expense Types audit report on your copy of the production during initial pass. The report lists all earnings code, deduction code, U.S. taxes, and Canadian taxes expenses accounts defined on Chart Dist HR and Chart Keys HR tables in prior releases.

See “Prepare Your Database,” Running Application Audits, Running Dynamic ChartFields Audits.

- Manually set up the GL activity groupings. Work in conjunction with general ledger staff to make sure you will group payroll activities and be able to map them correctly to corresponding ChartFields in a later step.

See “Complete Database Changes,” Setting Up General Ledger Interface, Setting Up the GL Activity Groupings.

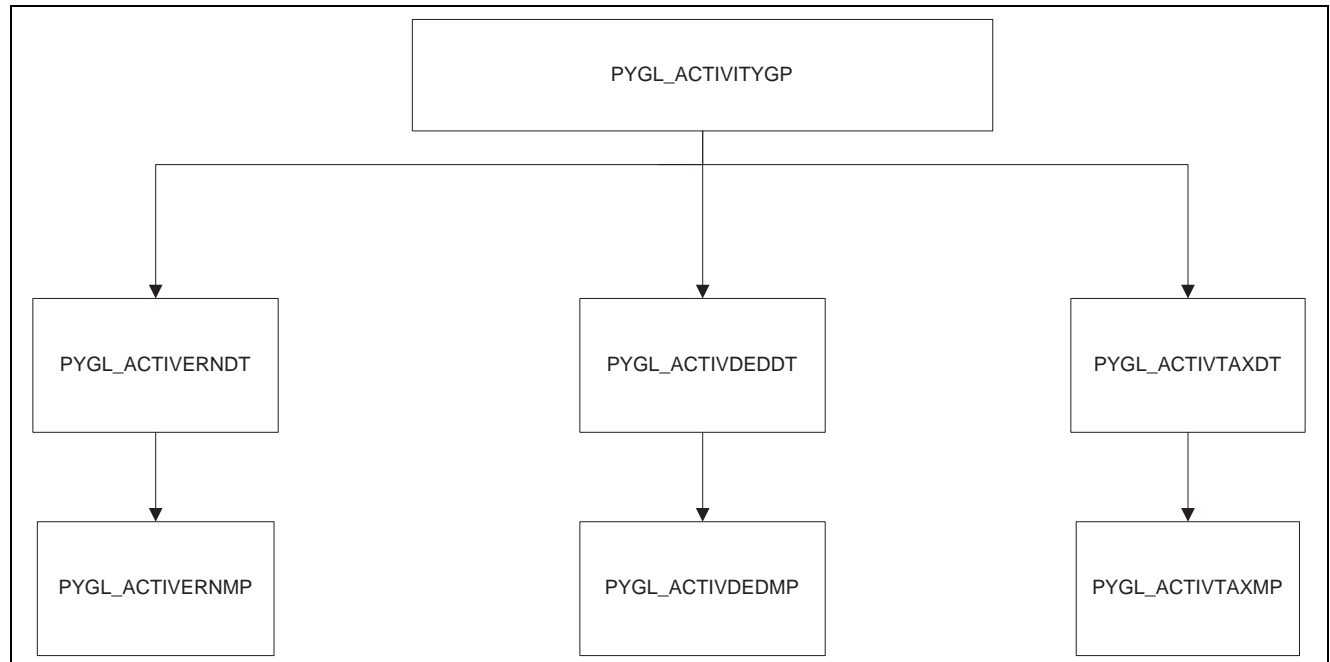
- Export tables updated in your copy of the production during initial pass of the upgrade.

See “Apply Application Changes,” Completing Application Processes, Exporting GL Interface Setup Tables.

- Import tables into your new copy of the production during Move to Production pass.

See “Apply Application Changes,” Completing Application Processes, Importing GL Interface Setup Tables.

Tables created for Payroll Activities grouping are:



Tables created for Payroll Activities Grouping

PYGL\_ACTIVITYGP table stores activity groups by Company. Earnings, Deductions, and Taxes Indicators can be checked ON in any combinations.

Field Name	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
DESCR	Activity Group description
PYGL_EARN_GRP	Yes / No indicator for Earnings
PYGL_DEDUCT_GRP	Yes / No indicator for Deductions
PYGL_TAX_GRP	Yes / No indicator for Taxes

PYGL\_ACTIVERNDT stores the effective date of each Earnings activity group.

Field Name	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
EFFDT	Effective date of the Earnings Group
PYGL_ALLCODES	Yes / No indicator to identify if this activity group includes All earnings codes. Note: For each Company, there should be only one Earnings Activity Group that has this indicator checked ON

PYGL\_ACTIVERNMP stores all earnings codes that are included in the Earnings activity group.

Field Name	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
EFFDT	Effective date of the Earnings Group
ERNCD	Earnings Code

Following is an example of the Earnings Activity Group setup for Sick Leave:

The screenshot displays the 'Earnings Expenses' tab in a software interface. It shows the following configuration:

- Company:** GBI
- Group Name:** SICK LEAVE
- Description:** Sick Leave Earn Codes
- Valid Activity Type(s):**
  - ☒ Earnings Group
  - ☐ Deduction Group
  - ☐ Tax Group
- Group Definition:**
  - Effective Date:** 01/01/2004
  - ☐ Group contains all codes
  - Codes in Group Definition:**

*Earnings Code	Description		
CSK	Sick Pay	+	-
KW5	Sick	+	-
SCK	Sick Leave	+	-
SIC	Hourly Sick Pay	+	-

Earnings Expenses page

PYGL\_ACTIVDEDDT stores the effective date of each Deduction activity group

FieldName	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
EFFDT	Effective date of the Deduction Group
PYGL_ALLCODES	Yes / No indicator to identify if this activity group includes All deduction codes. Note: For each Company, there should be only one Deduction Activity Group that has this indicator checked ON

PYGL\_ACTIVDEDMP stores all deduction codes that are included in the Deduction activity group.

FieldName	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
EFFDT	Effective date of the Deduction Group
PLAN_TYPE	Deductions plan type
DED_CLASS	Deduction Classification
DEDCD	Deduction Code
DED_SLSTX_CLASS	Sales Tax Type

Following is an example of the Deduction Activity Group setup for Health Benefits:

Earnings Expenses
Deduction Expenses
Tax Expenses

**Company** GBI  
**Group Name** HEALTH BENEFITS  
**Description** Health Benefits - Med / Dental

**Valid Activity Type(s)**  
☐ Earnings Group  
☒ Deduction Group  
☐ Tax Group

**Group Definition** Find | View All First 1 of 1 Last  
**Effective Date** 01/01/2004 ☐ Group contains all codes + -  
**Codes in Group Definition** Customize | Find | View All | First 1-5 of 6 Last

*Plan Type	*Ded Class	*Deduction Code	*Sales Tax Type	Description		
10	N	KUMED	B	Basic Domestic Partner Med		
10	N	KUMED9	B	Domestic Partner Medical		
10	T	KCMED	B	Basic Medical Deduction		
10	T	KCMED9	B	Medical Deductions		
11	N	KUDEN9	B	Dental Plan		

Deduction Expenses page

PYGL\_ACTIVTAXDT stores the effective date of each Tax activity group.

FieldName	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier

FieldName	Description
EFFDT	Effective date of the Tax Group
PYGL_ALLCODES	Yes / No indictor to identify if this activity group includes All taxes. Note: For each Company, there should be only one Tax Activity Group that has this indicator checked ON

PYGL\_ACTIVTAXMP stores all taxes that are included in the Tax activity group.

FieldName	Description
COMPANY	Company Code
PYGL_GROUPNAME	30 characters group identifier
EFFDT	Effective date of the Deduction Group
PYGL_TAX_COUNTRY	Either USA or Canada
STATE	State (for US only)
LOCALITY	Locality (for US only)
WAGE_LOSS_PLAN	Wage Loss Replacement Plan (for Canada only)
PYGL_TAX_CLASS	Tax Class activities

Following is an example of the Tax Activity Group setup for Federal FICA Tax:

**Earnings Expenses** **Deduction Expenses** **Tax Expenses**

**Company** GBI

**Group Name** FEDERAL FICA

**Description** Federal FICA Tax

**Valid Activity Type(s)**

- ☐ Earnings Group
- ☐ Deduction Group
- ☒ Tax Group

**Group Definition** Find | View All First 1 of 1 Last

**Effective Date** 01/01/2004 ☐ Group contains all codes

**Codes in Group Definition** Customize | Find | View All First 1 of 1 Last

*Country	State	Locality	Wage Loss Replacement Plan	Tax Class	Description
USA				FMR	Employer FICA Med Hospital Ins

Tax Expenses page example for FEDERAL FICA Tax

Following is an example of the Tax Activity Group setup for Canada CPP Tax:

**Earnings Expenses** **Deduction Expenses** **Tax Expenses**

**Company** GBI

**Group Name** COMPANY CPR

**Description** Company CPR

**Valid Activity Type(s)**

- ☐ Earnings Group
- ☐ Deduction Group
- ☒ Tax Group

**Group Definition** Find | View All First 1 of 1 Last

**Effective Date** 01/01/2000 ☐ Group contains all codes + -

**Codes in Group Definition** Customize | Find | View All First 1-2 of 2 Last

*Country	State	Locality	Wage Loss Replacement Plan	Tax Class	Description
Canada			KLO	CPR	Employer Canada Pension
Canada			KRG	CPR	Employer Canada Pension

Tax Expenses page example for Canada CPP Tax

## See Also

*PeopleSoft Enterprise Payroll for North America PeopleBook, for your new release.*

## Task L-2-3: Understanding Expenses ChartField Mapping

In the new PeopleSoft release, organizations that use non-commitment accounting, can assign ChartFields to payroll activity in various report levels. There are five report levels – Company, Department, Position, JobCode, and employment record. Organizations can track expenses down to individual employment record level if desired

Earnings, Deductions, and Taxes are defined as separate mapping entries. Therefore, they could be mapped to different levels. For example, organizations can track earnings expenses in employment record level and track fringe benefits (deductions and taxes) at Department level. The General Ledger Interface process retrieves these mappings starting from the lowest level (employment record) and reaches up to JobCode, Position, Department, and Company until a mapping definition is found.

If you are using non-commitment accounting you must do following:

1. Run and review Chartkey Usage by Company and Business Unit and Chartkey Usage by Expense Types audit reports on your copy of the production during initial pass.

The reports list all earnings code, deduction code, U.S. taxes, and Canadian taxes expenses accounts defined on Chart Dist HR and Chart Keys HR tables in prior releases.

See “Prepare Your Database,” Running Application Audits, Running Dynamic ChartFields Audits.

2. Manually set up the expenses ChartField mappings.

Work in conjunction with general ledger staff to make sure you will map payroll activities groupings correctly to each company/business unit combination.

See “Complete Database Changes,” Setting Up General Ledger Interface, Setting Up the Expenses ChartField Mappings.

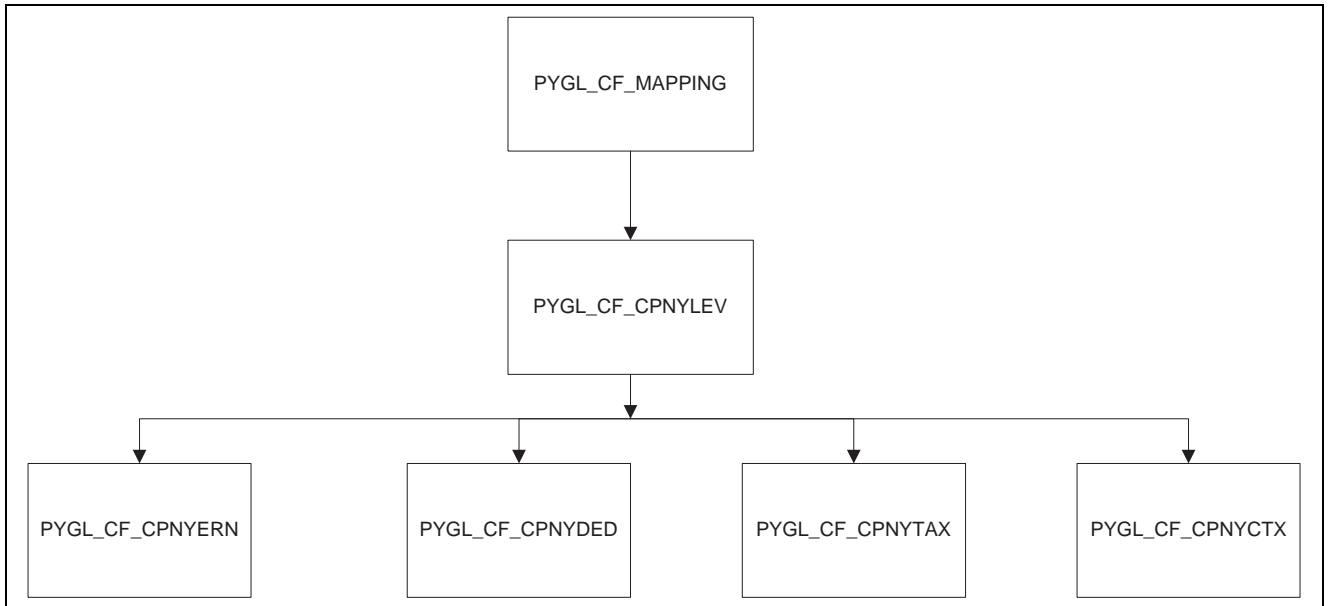
3. Export tables updated in your copy of the production during initial pass of the upgrade.

See “Apply Application Changes,” Completing Application Processes, Exporting GL Interface Setup Tables.

4. Import tables into your new copy of the production during Move to Production pass.

See “Apply Application Changes,” Completing Application Processes, Importing GL Interface Setup Tables.

Tables created for Company ChartField Mappings:



Tables created for Company ChartField Mappings

PYGL\_CF\_MAPPING table stores Company and HR Business Unit that require Expense Account setup.

FieldName	Description
COMPANY	Company Code
BUSINESS_UNIT	HR Business Unit

PYGL\_CF\_CPNYLEV table stores the level of the mapping.

FieldName	Description
COMPANY	Company Code.
BUSINESS_UNIT	HR Business Unit
DEPTID	Department ID
JOBCODE	Job Code
POSITION_NBR	Position Number
EMPLID	Employment ID
EMPL_RCD	Employment Record
EFFDT	Effective date of the mapping level

FieldName	Description
EFF_STATUS	Effective status of the mapping level
COMPANY_LEVEL_INDC	Mapping Level – either Company, Department, Position, JobCode or Employment record.

PYGL\_CF\_CPNYERN table stores ChartField assignment to each earnings activity group.

PYGL\_CF\_CPNYDED table stores ChartField assignment to each deduction activity group.

PYGL\_CF\_CPNYTAX table stores ChartField assignment to each US Tax activity group.

PYGL\_CF\_CPNYCTX table stores ChartField assignment to each Canada Tax activity group.

All these tables share the same record layout:

FieldName	Description
COMPANY	Company Code
BUSINESS_UNIT	HR Business Unit
DEPTID	Department ID
JOB CODE	Job Code
POSITION_NBR	Position Number
EMPLID	Employment ID
EMPL_RCD	Employment Record
EFFDT	Effective date of the mapping level
PYGL_GROUPNAME	Payroll Activity Group name
ACCT_CD	ChartField Combination code

Sections of audit reports listed under table name will help you in deciding how you want to set up data in each table:

- PYGL\_CF\_CPNYERN table

Review the expense accounts listed in:

- ChartKey Usage by Company and Business Unit audit report, Company Earnings Accounts section
- ChartKey Usage by Expense Type audit report, Earnings Expense Accounts section

- PYGL\_CF\_CPNYDED table

Review the expense accounts listed in:

- ChartKey Usage by Company and Business Unit audit report, Company Deduction Accounts section
- ChartKey Usage by Expense Type audit report, Deduction Expense Accounts section

- PYGL\_CF\_CPNYTAX table



Review the expense accounts listed in:

- ChartKey Usage by Company and Business Unit audit report, US Company Tax Accounts section
- ChartKey Usage by Expense Type audit report, US Tax Expense Accounts section
- PYGL\_CF\_CPNYCTX table

Review the expense accounts listed in:

- ChartKey Usage by Company and Business Unit audit report, Canadian Company Tax Accounts section
- ChartKey Usage by Expense Type audit report, Canadian Tax Expense Accounts section

Following is an example of the Earnings Mapping setup for Sick Leave at Employee Level:

Earnings Mapping		Deductions Mapping		U.S. Tax Mapping		Canadian Tax Mapping	
<b>Company:</b>	GBI	Global Business Institute 9999					
<b>Business Unit:</b>	US004	GBI BU for US004					
<b>Expense ChartField Level</b>							
<input checked="" type="radio"/> Company	<input type="radio"/> Department	<input type="radio"/> Jobcode	<input type="radio"/> Position	<input type="radio"/> Employee			
<b>EmplID:</b>	KU0010	Antonio Santos					
<b>Empl Rcd Nbr:</b>	0						
<b>Effective Date:</b>	01/01/2004	<b>Status:</b>	Active				
<b>Chartfields</b>							
	<u>*GL Group Name</u>	<u>Combination Code</u>	<u>Edit ChartFields</u>	<u>Account</u>			
1	SICK LEAVE	610000-1	Edit ChartFields	610000			

Earnings Mapping page; Setup for Sick Leave at Employee Level

Following is an example of the Earnings Mapping setup for Sick Leave at Department Level:

Earnings Mapping		Deductions Mapping		U.S. Tax Mapping		Canadian Tax Mapping	
<b>Company:</b>	GBI	Global Business Institute 9999					
<b>Business Unit:</b>	US004	GBI BU for US004					
<b>Expense ChartField Level</b>							
<input type="radio"/> Company	<input checked="" type="radio"/> Department	<input type="radio"/> Jobcode	<input type="radio"/> Position	<input type="radio"/> Employee			
<b>Department</b>	10000	Human Resources					
<b>Effective Date:</b>	01/01/2004	<b>Status:</b>	Active				
<b>Chartfields</b>							
	<u>*GL Group Name</u>	<u>Combination Code</u>	<u>Edit ChartFields</u>	<u>Account</u>			
1	SICK LEAVE	610000-2	Edit ChartFields	610000			

Earnings Mapping Tab; Setup for Sick Leave at Department Level

Following is an example of the Deduction Mapping setup for Health Benefits at Department Level:

Earnings Mapping Deductions Mapping U.S. Tax Mapping Canadian Tax Mapping

**Company:** GBI Global Business Institute 9999  
**Business Unit:** US004 GBI BU for US004

**Expense ChartField Level**

☐ Company ☒ Department ☐ Jobcode ☐ Position ☐ Employee + -

**Department** 10000 Human Resources  
**Effective Date:** 01/01/2004 31 **Status:** Active ▼

**Chartfields**

	*GL Group Name	Combination Code	Edit ChartFields	Account
1	HEALTH BENEFITS	P00000003	Edit ChartFields	610000

Deductions Mapping tab; Setup for Health Benefits at Department Level

Following is an example of the US Tax Mapping setup for Federal FICA Tax at Department Level:

Earnings Mapping Deductions Mapping U.S. Tax Mapping Canadian Tax Mapping

**Company:** GBI Global Business Institute 9999  
**Business Unit:** US004 GBI BU for US004

**Expense ChartField Level**

☐ Company ☒ Department ☐ Jobcode ☐ Position ☐ Employee + -

**Department** 10000 Human Resources  
**Effective Date:** 01/01/2004 31 **Status:** Active ▼

**Chartfields**

	*GL Group Name	Combination Code	Edit ChartFields	Account
1	FEDERAL FICA	615000-5	Edit ChartFields	615000

US Tax Mapping tab; Setup for Federal FICA Tax at Department Level

Following is an example of the Canada Tax Mapping setup for CPP Tax at Department Level:

Earnings Mapping Deductions Mapping U.S. Tax Mapping Canadian Tax Mapping

**Company:** GBI Global Business Institute 9999  
**Business Unit:** US004 GBI BU for US004

**Expense ChartField Level**

☐ Company ☒ Department ☐ Jobcode ☐ Position ☐ Employee + -

**Department** 10000 Human Resources  
**Effective Date:** 01/01/2004 31 **Status:** Active ▼

**Chartfields**

	*GL Group Name	Combination Code	Edit ChartFields	Account
1	COMPANY CPR	615000-2	Edit ChartFields	615000

Canadian Tax Mapping tab; Setup for CPP Tax at Department Level

## See Also

*PeopleSoft Enterprise Payroll for North America PeopleBook, for your new release.*

---

## Task L-3: Upgrading Time and Labor ChartFields

ChartField enhancements are provided with the new PeopleSoft release. The individual ChartFields were expanded from 14 fields to 16 fields and the Account Code field was relabeled as Combo Code. The biggest change is that now commitment accounting and non-commitment accounting customers have access to all ChartFields and the associated functionality. A distinction between the two no longer exists in Time and Labor.

Prior to the new release, a user was able to report an Account Code and a Project ID for a single transaction. The reported Account Code might have an associated Project ID. The associated Project ID could be the same or different from the Project ID entered on the transaction row. This functionality is still available in the new release, however; if Project ID is an active ChartField then the Combo Code that is reflected on the transaction needs to represent the combination of associated individual ChartField values correctly.

To upgrade the data from prior releases correctly, a process was created to identify all transactions that have an Account Code and Project ID reported together. Data from TL\_PAYABLE\_TIME, TL\_RPTD\_TIME and TL\_TSKPRF\_DETL tables is pulled into an upgrade table, and then the comparison is performed to identify if the Project ID associated with the Account Code is the same as the Project ID reported directly. If the Account Code does not reflect the reported Project ID, then a new Combo Code is generated or an existing Combo Code is identified, from the ACCT\_CD\_TBL, and all transaction rows associated are updated with the new value. This process will only impact transaction rows where the application is integrated with PeopleSoft Project Costing and not transaction rows that have a Time and Labor Project reported.

During the process outlined above, Combo Codes that were inactive at the start of the upgrade may be reactivated. The new release captured these Combo Codes for informational purposes. The following query: 'UPG\_TL\_ACCT\_CD\_ACTIVATED – Upgrade TL activated Account Codes' displays a list of impacted Combo Codes. If you are using the Account Code Load Process from PeopleSoft Financials, run this at the end of the upgrade to inactivate any Combo Codes that are no longer valid.

In the new release, the Project ID associated with a Combo Code is stored in the project field throughout the application. Prior to the new release, only the Account Code was stored, the associated Project ID was not. In the new release, this will cause exceptions to be generated when Interface with PeopleSoft/Projects flag is checked on the Time and Labor Installation page. To assist with preventing these exceptions, the upgrade process will query all transaction rows to determine if a reported Account Code has an associated Project ID and a Business Unit PC also exists. For these transactions the Project ID associated with the Combo Code will be stored in the project field.

The new release provides some queries that can be generated to compare the upgraded data to what existed in the system prior to the upgrade process. These queries all begin with UPG\_TL and are:

- UPG\_TL\_CFOLDNEW – Upgrade TL Account Code  
Provides a list of account codes that have been impacted during the upgrade. They are either old ones that need to be reactivated or new ones that have been generated.
- UPG\_TL\_PAY\_NEWACC – TL Upgrade Payable Time New Account Code  
Provides a list of payable time rows that have been changed. The existing account code has been replaced with a newly generated account code that represents the combination of data originally stored in payable time.
- UPG\_TL\_PAY\_REUSED – TL Upgrade Payable Time Reused Account Code

Provides a list of payable time rows that have been changed. The existing account code has been replaced with a previously existing account code that represents the combination of data originally stored in payable time.

- UPG\_TL\_PAY\_UPDATED – TL Upgrade Payable Time Updated

Provides a list of payable time rows that have been changed because the Project ID associated with the account code is now stored in the project id field.

- UPG\_TL\_PRJ\_NEWACC – TL Upgrade Task Profile New Account Code

Provides a list of task profiles that have been changed. The existing account code has been replaced with a newly generated account code that represents the combination of data originally stored on the task profile.

- UPG\_TL\_PRJ\_REUSED – TL Upgrade Task Profile Reused Account Code

Provides a list of task profiles rows that have been changed. The existing account code has been replaced with a previously existing account code that represents the combination of data originally stored on the task profile.

- UPG\_TL\_PRJ\_UPDATED – TL Upgrade Task Profile Updated

Provides a list of task profiles that have been changed because the Project ID associated with the account code has been stored in the project id field.

- UPG\_TL\_RPTD\_NEWACC – TL Upgrade Reported Time New Account Code

Provides a list of reported time rows that have been changed. The existing account code has been replaced with a newly generated account code that represents the combination of data originally stored in reported time.

- UPG\_TL\_RPTD\_REUSED – TL Upgrade Reported Time Reused Account Code

Provides a list of reported time rows that have been changed. The existing account code has been replaced with a previously existing account code that represents the combination of data originally stored in reported time.

- UPG\_TL\_RPTD\_UPDATED – TL Upgrade Reported Time Updated

Provides a list of reported time rows that have been changed because the Project ID associated with the account code is now stored in the project id field.

Additionally, the new release provides a query for users to identify if the Combo Code associated to a Task Profile is invalid, 'UPG\_TL\_INVALID\_TSKPRF – Task Profile-Invalid Combo Code'. This query will only validate using the table method and not the sync message with PeopleSoft Financials. Run this anytime individual ChartField values or Combo Codes are inactivated.

## APPENDIX M

# Understanding FTE Enhancement

---

## Understanding FTE Enhancement

This appendix explains the enhancements made in the new PeopleSoft release for the Full Time Equivalence (FTE) and the Full Time Equivalence-related calculations.

To improve precision in FTE and FTE-related calculations, such as employee compensation, PeopleSoft has expanded the number of decimal positions displayed and calculated in the FTE field to six.

In the new PeopleSoft release, the FTE field has the following definition: 1 integer, 6 decimals.

In addition, a set of external rounding rules (Set Up HRMS, Foundation Tables, Organization, Company - Rules Definition tab) has been provided that allow you to define at the company level the preferred number of significant decimals for FTE calculations. A new table PS\_COMPANY\_TBL\_AR was created for housing the effective-dated rounding rules.

The new upgrade process creates a new FTE rounding rule of two-decimal-precision for each of your existing companies with the lowest effective date available for the particular company. By doing this, the rounding rule is matched with FTE decimal size from the previous release. (In previous HRMS releases, FTE field had the following definition: 1 integer, 2 decimals.)

Therefore, after you have completed your upgrade to the new PeopleSoft release, it is recommended you visit company table set up and double check the current value for your existing companies. If you had customized FTE decimal size in your system, you may want to modify FTE rounding rule values to match your customized FTE field definition.

---

**Note.** FTE enhancements extend to the French-specific PAID FTE field as well.

---



## APPENDIX N

# Understanding Garnishments Changes

This appendix discusses:

- Understanding Garnishments Upgrade
- Understanding Payee Changes
- Understanding Rules Changes

---

## Understanding Garnishments Upgrade

In PeopleSoft 9, we enhanced Garnishments functionality, which impacted the following areas:

- Garnishment Payee Data
- Garnishment Rule Data

We added the Location field to the payee data we store on the garnishment order to enhance processes that create invoices for paid orders. You will notice this change when you are creating new garnishment orders, whether you are using U.S. Payroll or Canadian Payroll. However, the upgrade related to this change is mandatory only if you are using our AP Interface functionality to create invoices.

We made several changes in the way we store U.S. garnishments rules, which you will notice if you are using U.S. Payroll or Federal Payroll to set up or review garnishments rules. There is no change in the way that we store Canadian garnishments rules. However, due to the fact that garnishments rules tables store data for U.S. and Canadian rules, and may store rules maintained by you in addition to the rules we deliver, even if you only use Canadian Payroll, your garnishments data will be processed by some upgrade tasks. This was the only way for us to deliver a generic upgrade that will work for any combination of data a user might have.

---

## Task N-1: Understanding Payee Changes

This section discusses:

- Understanding Changes to Payee Data
- Updating Payee Data During the Initial Pass
- Updating Payee Data During Move to Production

## Understanding Changes to Payee Data

If you are using AP Interface to create an invoice for the payee, you have to store payee data in the garnishment order. The payee data must exist in the Payee Table in order to be entered into the garnishment order. When you enter the payee data, the payee's address details will be added to the garnishment order, but it will not be used in any further processing. When the invoice and the voucher are created, the data sent to Accounts Payable does not contain the details about the payee's address; instead, it only has key information from the Payee Table that will allow AP processes to find the details on their side. This processing is based on the assumption that the Payee Table is synchronized on both sides, and it was designed in such way to avoid sending unnecessary information.

In PeopleSoft 9, we have added the Location field to the garnishment order (GARN\_SPEC record) to ensure that we will send the correct keys to the AP side, which will in return find the correct address details when sending vouchers to the payees.

During the conversion, we will update the location on the garnishment order with the default location from the Payee Table. However, in some cases that might not be enough to complete the upgrade correctly. The following scenarios are possible:

- Some of your payees might be set up with different invoicing and remitting VendorIDs.
- Some of your payees might have multiple locations.

If you have any payees with either of the above setup scenarios, we will need your input to complete the upgrade.

Following are simplified tasks related to garnishment payee data, which you can perform during the upgrade.

### Task N-1-1: Updating Payee Data During the Initial Pass

To update garnishment payee data during the Initial pass:

1. Run the Garnishment Payee Audit Report.
2. Review the report listing and fix any garnishment order or payee setup for the payees that are referenced on the garnishment order, but do not exist in the Payee Table.

See "Prepare Your Database," Running Application Audits, Correcting Garnishment Payee Data, Running the Garnishment Payee Audit Report.

3. Run conversion for the Garn Spec record.

The conversion will perform the following updates:

- Update the Location field on the Garn Spec record with the default location from the Payee Table.
- Update the Location and Remit Address on the Garn Spec field with your input, which in this pass will not be available; therefore, there will be no updates.

See the comments on the Upgrade Data Conversion page. Select HRMS, Upgrade, Data Conversion, and then select the Comments link for Update GARN\_SPEC.

4. Run the EE Garn Payee Data Validation Report.
5. Review the report listing and decide what data needs your input to complete the conversion.

See "Complete Database Changes," Validating EE Garn Payee Data.

6. Add your input for updating Garn Payee Data where needed.

---

**Note.** These entries will be used in the subsequent pass during the conversion to update Garn Spec records.

---



See “Complete Database Changes,” Validating EE Garn Payee Data.

## Task N-1-2: Updating Payee Data During Move to Production

To update garnishment payee data during the Move to Production pass:

1. Run the Garnishment Payee Audit Report.
2. Review the report listing and fix any garnishment order or payee setup for the payees that are referenced on the garnishment order, but do not exist in the Payee Table.

See “Prepare Your Database,” Running Application Audits, Correcting Garnishment Payee Data, Running the Garnishment Payee Audit Report.

3. Run conversion for the Garn Spec record.

The conversion will perform the following updates:

- Update the Location field on Garn Spec record with the default location from the Payee Table.
- Update Location and Remit Address on the Garn Spec field with your input, which in this pass will be available.

See the comments on the Upgrade Data Conversion page. Select HRMS, Upgrade, Data Conversion, and then select the Comments link for Update GARN\_SPEC.

4. Run the EE Garn Payee Data Validation Report.
5. Review the report listing and decide what data needs your input to complete the conversion. This report may contain data if you have created a new garnishment order between upgrade passes with a payee that does not exist on the Payee Table.

See “Complete Database Changes,” Validating EE Garn Payee Data.

6. Add your input for updating Garn Payee Data where needed.

---

**Note.** These entries will be used in subsequent Move to Production passes during the conversion to update Garn Spec records. If this is the final Move to Production pass, you need to do manual updates directly in the Create Garnishment or Payee Table component because there will be no more conversions to take care of these entries.

---

See “Complete Database Changes,” Validating EE Garn Payee Data.

---

## Task N-2: Understanding Rules Changes

This section discusses:

- Understanding Changes to Garnishment Rules
- Updating Rules During the Initial Pass
- Updating Rules During Move to Production

## Understanding Changes to Garnishment Rules

In PeopleSoft 9, we have changed the way that we store U.S. garnishment rules data. Even though there is no change to the way that we store Canadian garnishment rules data, due to the fact that all data is stored in the same set of setup tables, there will be some upgrade processing on your garnishments rules tables.

The garnishment rules tables may have the following rules:

- U. S. rules maintained and delivered by Oracle
- Canadian rules maintained and delivered by Oracle
- U.S. rules created and maintained by the user
- Canadian rules created and maintained by the user

You may not have all of the above data. You probably have the rules delivered by Oracle, and in addition, you might have some custom rules for either payroll. In order to ensure that the upgrade processing will work for any combination of data you may have, we decided to do the following conversion on garnishments rules tables:

- Rename garnishments rules tables in your Copy of Production.
- Export U.S. and Canadian rules maintained by Oracle from the New Release Demo database.  
There is no change to Canadian rules maintained by us, but redelivering them was a better option than converting them from the renamed tables.
- Import U.S. and Canadian rules maintained by Oracle into your Copy of Production before the conversion.
- Copy Canadian custom garnishment rules, if you currently have any, from renamed tables into garnishments rules tables in your Copy of Production.

We are not going to do any conversion for the U.S. custom garnishment rules, if you currently have any. The changes that we made in garnishment rules tables would require a very complex upgrade with a high probability of creating invalid data in your tables. On the other hand, the flexibility in setting up the rules might allow you to replace your existing custom rule with a rule delivered by Oracle. Depending on whether you decide that you still need the U.S. custom rules or not, you may need to recreate the U.S. custom rules, or you may need to update garnishment orders to replace a U.S. custom rule with some other existing rule.

Following are simplified tasks related to rules changes, which you can perform during the upgrade:

### Task N-2-1: Updating Rules During the Initial Pass

To update garnishment rules during the Initial pass:

1. Run the US Custom Garnishment Rule Audit Report.
2. Review the report listing and decide whether you need any of these rules in PeopleSoft 9.

See “Prepare Your Database,” Running Application Audits, Running Upgrade Reports, Running US Custom Garnishment Rule Audit Report.

3. Run conversion for Garn Rule record (GARN\_RULE).

The conversion will update the State Law and Rule ID fields with your input, which in this pass will not be available; therefore, there will be no updates.

See the comments on the Upgrade Data Conversion page. Select HRMS, Upgrade, Data Conversion, and then select the Comments link for Update GARN\_RULE.

4. Set up US Custom Garnishment Rules.

If you have any U.S. custom garnishments rules that you will need in new release, you will have to re-create them. You can set up the rule using the online component or you can clone any existing rule if the formula is similar to the rule you need to create. You can use the audit report to ensure that all data is set up the way you need it.

See “Complete Database Changes,” Setting Up US Custom Garnishment Rules.

5. Run the EE Garn Rule Data Validation Report.
6. Review the report listing and decide what data needs your input to complete the conversion, in case you want to replace an obsolete rule with an existing rule.

See “Complete Database Changes,” Validating EE Garn Rule Data.

7. Add your input for updating rule data in garnishment orders, where needed.

---

**Note.** These entries will be used in the following pass during the conversion to update Garn Rule records.

---

See “Complete Database Changes,” Validating EE Garn Rule Data.

## Task N-2-2: Updating Rules During Move to Production

To update rules during the Move to Production pass:

1. Run the US Custom Garnishment Rule Audit Report.
2. Review the report listing and decide whether you need any of these rules in PeopleSoft 9. Compare the report from the current and previous passes to see whether any new rules were added between upgrade passes.

See “Prepare Your Database,” Running Application Audits, Running Upgrade Reports, Running US Custom Garnishment Rule Audit Report.

3. Run conversion for the Garn Rule record (GARN\_RULE).

The conversion will update the State Law and Rule ID fields with your input, which will be available in this pass if you added any.

See the comments on the Upgrade Data Conversion page. Select HRMS, Upgrade, Data Conversion, and then select the Comments link for Update GARN\_RULE.

4. Set up US Custom Garnishment Rules.

If you have any new U.S. custom garnishments rules that you have set up between upgrade passes, and you have decided that you will need them in the new release, you must recreate them. You will have to do set up only for the U.S. rules newly added between the upgrade passes. All other U.S. rules set up in previous passes, will be taken care of by running the export/import scripts. You can set up the rule using the online component or you can clone any existing rule if the formula is similar to the rule you need to create. You can use the audit report to ensure that all data is set the way you need it.

See “Complete Database Changes,” Setting Up US Custom Garnishment Rules.

5. Run the Export US Custom Garnishments Rules script.

Run this script to export from your Copy of Production all U.S. custom garnishments rules that you have set up manually in the previous pass.

See “Apply Application Changes,” Completing Application Processes, Exporting US Custom Garnishment Rules.

6. Run the Import US Custom Garnishments Rules script.

Run above script to import into your new Copy of Production all U.S. custom garnishments rules that you have set up manually in the previous pass.

See “Apply Application Changes,” Completing Application Processes, Importing US Custom Garnishment Rules.

7. Run the EE Garn Rule Data Validation Report.
8. Review the report listing and decide what data needs your input to complete the conversion.

This report may contain data if you have created a new garnishment order in between upgrade passes that references a rule that no longer exists in garnishment rules tables.

See “Complete Database Changes,” Validating EE Garn Rule Data.

9. Add your input for updating rule data in garnishment orders, where needed.

---

**Note.** These entries will be used in subsequent Move to Production passes during the conversion to update Garn Rule records. If this is the final Move to Production pass, you need to do manual updates directly in the Create Garnishment or Garnishment Rules Tables component, because there will be no more conversions to take care of these entries.

---

See “Complete Database Changes,” Validating EE Garn Rule Data.

See PeopleSoft Payroll for North America PeopleBook, for your new release.

## APPENDIX O

# Understanding Performance Management

This appendix discusses:

- Understanding Performance Management Changes
- Understanding Employee Review Tables
- Understanding Employee Review Data

---

## Understanding Performance Management Changes

This appendix describes the changes that have been made for ePerformance in the new PeopleSoft release.

---

**Note.** In the discussions that follow, the term “ePerformance” is used as a convenience to refer not only to the new product of that name, but also to any existing business functions contained in Career Planning, Salary Planning, and eDevelopment that are now implemented on top of these new data structures and business processes.

---

---

## Task O-1: Understanding Employee Review Tables

This section discusses:

- Understanding Employee Review Tables Changes
- Understanding Employee Review Records
- Understanding Performance Improvement Plan (USF) Records
- Understanding Merit Scale Setup Records
- Understanding Rating Scale Setup Records
- Understanding Review Band Setup Records
- Understanding Performance Plan (USF) Setup Records

## Understanding Employee Review Tables Changes

The ePerformance application was developed to provide a comprehensive but flexible tool for managing workforce performance toward a set of goals and objectives. Prior to the new PeopleSoft release, there was limited Employee Review functionality under the Salary Planning, Career Planning, and eDevelopment modules and was primarily limited to storing information about an employee’s performance against a set of competencies. It did not address the overall business process of hierarchical planning, collaborative goal setting and performance monitoring, and approval.

## Task O-1-1: Understanding Employee Review Records

The following are Employee Review Records:

REVIEW\_REVIEWER  
REVW\_REVWER\_CMP  
REVIEW\_GOALS  
REVIEW\_COMMENTS  
GVT\_EMP\_ELEMNT  
GVT\_EMP\_STAN  
GVT\_LATE\_RV\_TBL

## Task O-1-2: Understanding Performance Improvement Plan (USF) Records

The following are Performance Improvement Plan (USF) Records:

GVT\_EMPL\_PIP

## Task O-1-3: Understanding Merit Scale Setup Records

The following are Merit Scale Setup Records:

MERIT\_SCALE\_TBL  
MERIT\_GROUP\_TBL

## Task O-1-4: Understanding Rating Scale Setup Records

The following are Rating Scale Setup Records:

REVW\_SCALE\_TBL  
REVW\_FACTOR\_TBL  
REVW\_FCT\_PT\_TBL

## Task O-1-5: Understanding Review Band Setup Records

The following is a Review Band Setup Record:

REVIEW\_BAND\_TBL

## Task O-1-6: Understanding Performance Plan (USF) Setup Records

The following are Performance Plan Setup records:

GVT\_PERF\_PLAN  
GVT\_PERF\_ELEMNT  
GVT\_PERF\_STAN

As discussed below, the data in these tables will not be upgraded but the tables will temporarily be retained for historical reference. However, no editing or processing of this data will be available. This data can still be accessed in the following ways:

- **Ad Hoc Querying:** All of these tables are included in the ePerformance Query Access Tree and are therefore available for ad hoc querying.
- **Portal Access:** Navigate to Compensation, Salary Planning, Employee Review History. This includes Employee Review History, the Employee Review Audit (PER088), Review History USF, and Review Performance Improvement USF (both US Federal Government pages were previously under the Manage Performance USF menu but have been consolidated here). Note that these pages are read-only (no data can be changed) and no standard page access will be available for the underlying support tables.

These objects (records, pages and reports) will be retained (but no longer *maintained* functionally) for a reasonable period of time (one to two major releases) to allow for transition, at which point they will be dropped. If you want to retain them beyond that point, they will have to be treated like custom objects.

---

**Note.** U.S. Federal Government users: The field GVT\_PERF\_PLAN is no longer displayed on the Job Code Table or Position Data Table, since this field no longer applies to the new business process. The field itself is retained on these tables, however, so existing data can still be accessed through PS Query.

---

## Task O-2: Understanding Employee Review Data

In the new PeopleSoft release, Employee Review Data will not be upgraded. For ePerformance, a completely new, and highly flexible, record structure for managing employee reviews/appraisals (PS\_EP\_APPR and its children) has been developed. This involves the creation of template building blocks that define the content and behavior of any area of a review, such as mission objectives, initiatives, goals, responsibilities, competencies, and so forth. Reviews are keyed by a single unique Appraisal ID with alternate keys of Employee ID, Review Period (begin and end). The conducting of reviews during the review period is managed by a clear set of review statuses and a separate approval status. Different review types can be optionally marked as official. It is anticipated that the major use of this flag will be to indicate which reviews affect salary (as opposed to ad hoc reviews, or checkpoint reviews such as quarterly versus annual).

The existing Employee Review family (PS\_EMPLOYEE\_REVIEW) stores both the Commercial and Federal reviews/appraisals. After analyzing the key structure and functionality of the Employee Review structures, it was realized that there is no way to upgrade this data into the new ePerformance (PS\_EP\_APPR) key structure with any certainty of functional accuracy. This is because the keys to PS\_EMPLOYEE\_REVIEW are:

Primary Key: EMPLID, EMPL\_RCD, EFFDT

Optional Attributes: REVIEW\_TYPE, REVIEW\_FROM\_DT, and REVIEW\_THRU\_DT

While the keys to the new PS\_EP\_APPR structure are:

Primary Key: EP\_APPRAISAL\_ID

Alternate Key: EMPLID, EMPL\_RCD, PERIOD\_BEGIN\_DT, PERIOD\_END\_DT

The fact that PS\_EMPLOYEE\_REVIEW is effective-dated, and its FROM/THRU dates attributes are both *optional*, make it almost impossible to translate into PS\_EP\_APPR's required Period dates. Even the fundamental question of "What does a new effective-date mean?" is ambiguous. It could mean a new review period was started, or it could simply indicate that a checkpoint was taken. The tables' key structures do not prohibit a new effective-dated row from having a different review type with the same FROM/THRU dates as another row, or from having different or overlapping FROM/THRU dates, or from having no FROM/THRU dates at all. Further, the existing REVIEW\_TYPE translate values are a mixture of review *types* and review *statuses*. There is a REVIEW\_TYPE of Official Rating, but there is no other indication or status for *when* a review is "complete" and therefore whether its Rating has indeed become official.

In the final analysis, there is no benefit of having historical review data (which was conducted on top of an entirely different rating system and business process) brought into the new structure. Once a review is complete, it is not subject to modification (almost by definition). The main requirement for historical reviews is that they be available for query. (For the Federal Government, the requirement that the final rating for certain historical appraisals be available to the WGI and other processes is satisfied by changes made to certain program interfaces to consider both pre- and post-HRMS 9.0 reviews).



## APPENDIX P

# Understanding Person Model Changes

This appendix discusses:

- Understanding Changes
- Understanding Entity Relationships
- Understanding Record and Table Changes

---

## Understanding Changes

This appendix discusses the changes to the core record structures for a Person, the Person's basic biographical data, and the relationships that the Person has with the organization. These record structures evolved in each of the releases from PeopleSoft 8.0 to PeopleSoft 9.0. Because the biggest changes occurred between PeopleSoft 8.0 and PeopleSoft 8.3 and between PeopleSoft 8.3 and PeopleSoft 8.9, these changes will be the focus of this appendix.

The main goals of the changes were to:

1. Normalize the Person data structures as follows:
  - Remove the storing of a single address in 7 different tables.
  - Remove all redundant fields from PERSONAL\_DATA.
  - Simplify the maintenance of names and addresses.
2. Separate the Person data structures into components by moving non-core data into distinct records.
  - This was done to minimize the amount of blank fields that had to be captured in a record. If a customer doesn't use a particular country, then no data has to be stored in those fields.
  - While this does increase the number of records in PERSONAL\_DATA component, it allows us to separate the business logic and the processing. Eventually it will allow us to compartmentalize the actual component and only bring into the buffer the data that you need.
  - It also makes adding additional non-core fields easier to manage, as their addition will not break any usage of the core records.

3. Keep one ID for a person.

This feature adds the ability to have a single ID for a Person regardless of how many different relationships to the organization that person has. This means that you no longer have to create a separate ID if someone was an employee and is now a contingent worker.

4. Allow a Person to be created without having to create JOB records.

This change means that instead of having to create new EMPLIDs whenever a Person is both an Employee and a Non-employee, this model will allow a Person to have multiple organizational relationships. In

addition, we are splitting the Non-employees into two categories: Contingent Workers and Persons of Interest. This allows us to be able to treat Employees and Contingent Workers as part of the workforce — and the other Persons of Interest as not part of the workforce.

The three organizational relationships that a Person can have are:

- Employee (EMP)
- Contingent Worker (CWR)
- Person of Interest (POI)

Employees and Contingent Workers comprise the workforce and are required to have a JOB history.

Persons of Interest might have a JOB record if certain products process them. POIs include Cobra Participants, Pension Payees, and Stock non-emps, Learning Solution Students needing a refund via NA Payroll, External Trainers, and External Students. Because processes are already built for these groupings that need information on the JOB record, those groups will need to have a JOB record. Currently, the only POI types that don't need a JOB record are the External Trainers and External Students (including the Applicants needing training prior to hire).

The new model will allow a Person to be an Employee, a Contingent Worker, and/or a Person of Interest simultaneously or in succession using the same EMPLID. Each organizational relationship will be separated by EMPL\_RCD — meaning any one EMPL\_RCD for a Person can only be for one organizational relationship history. But there is no intelligence in the actual EMPL\_RCD number.

The PeopleSoft 8.81 to 8.9 upgrade deals with creating the new structure that allows a single EMPLID to have more than one Relationship to the organization. In addition, the separation of the non-core data is finished.

The main changes:

- The upgrade process will not attempt to merge EMPLIDs.
- The current EMPL\_RCDs assigned to an EMPLID will not change.
- No EMPL\_RCDs will be deleted.
- Customers do not have to remove references to PERSONAL\_DATA or EMPLOYMENT in their customized code.

---

## Task P-1: Understanding Entity Relationships

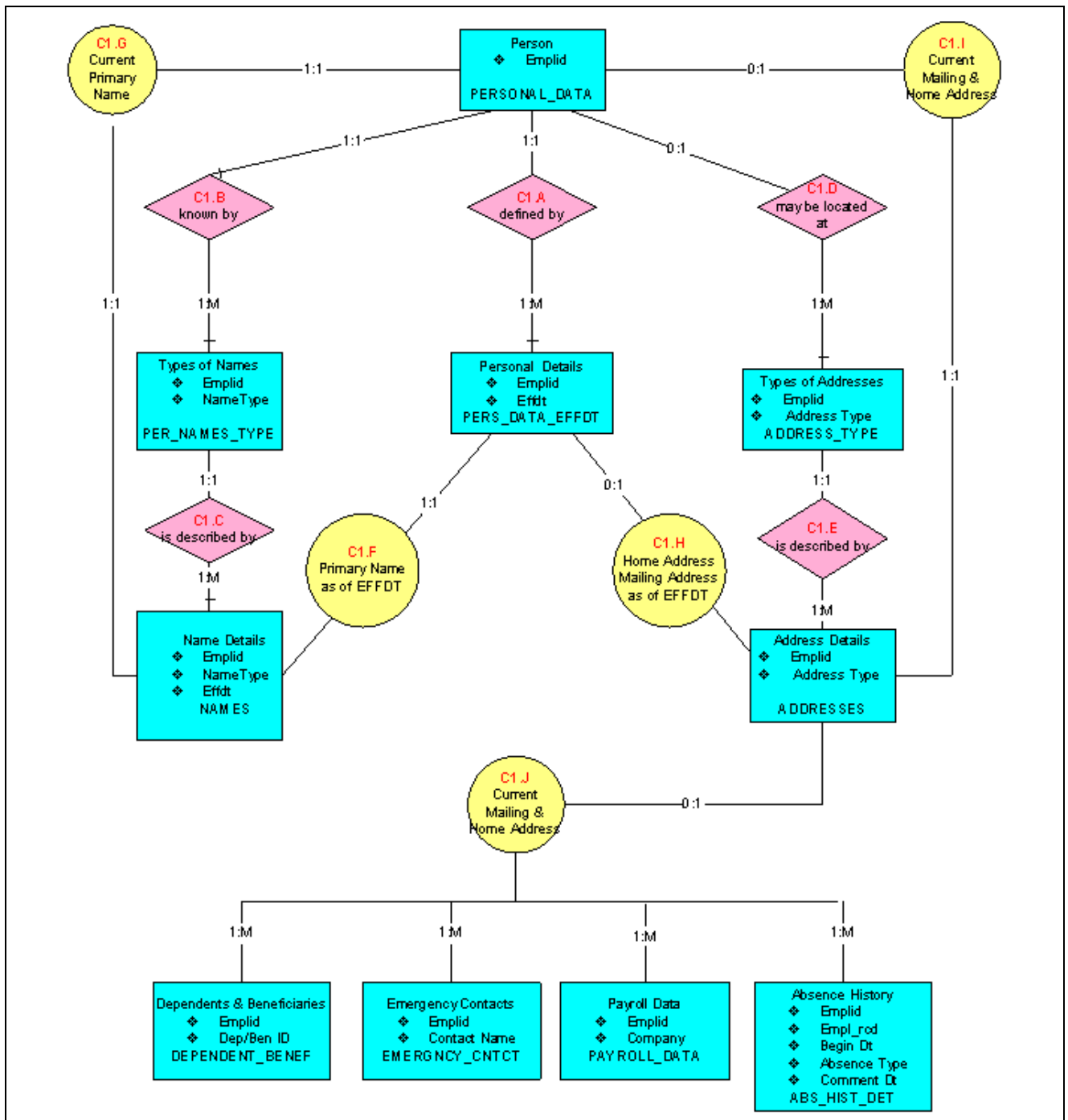
This section discusses:

- Understanding PeopleSoft 8.0 Entity Relationships
- Understanding PeopleSoft 8.8 Entity Relationships
- Understanding PeopleSoft New Release Entity Relationships

The following Entity Relationship Diagrams (ERDs) are designed to provide you with a visual representation of some of the key relationships Person records had in earlier PeopleSoft releases.

### Task P-1-1: Understanding PeopleSoft 8.0 Entity Relationships

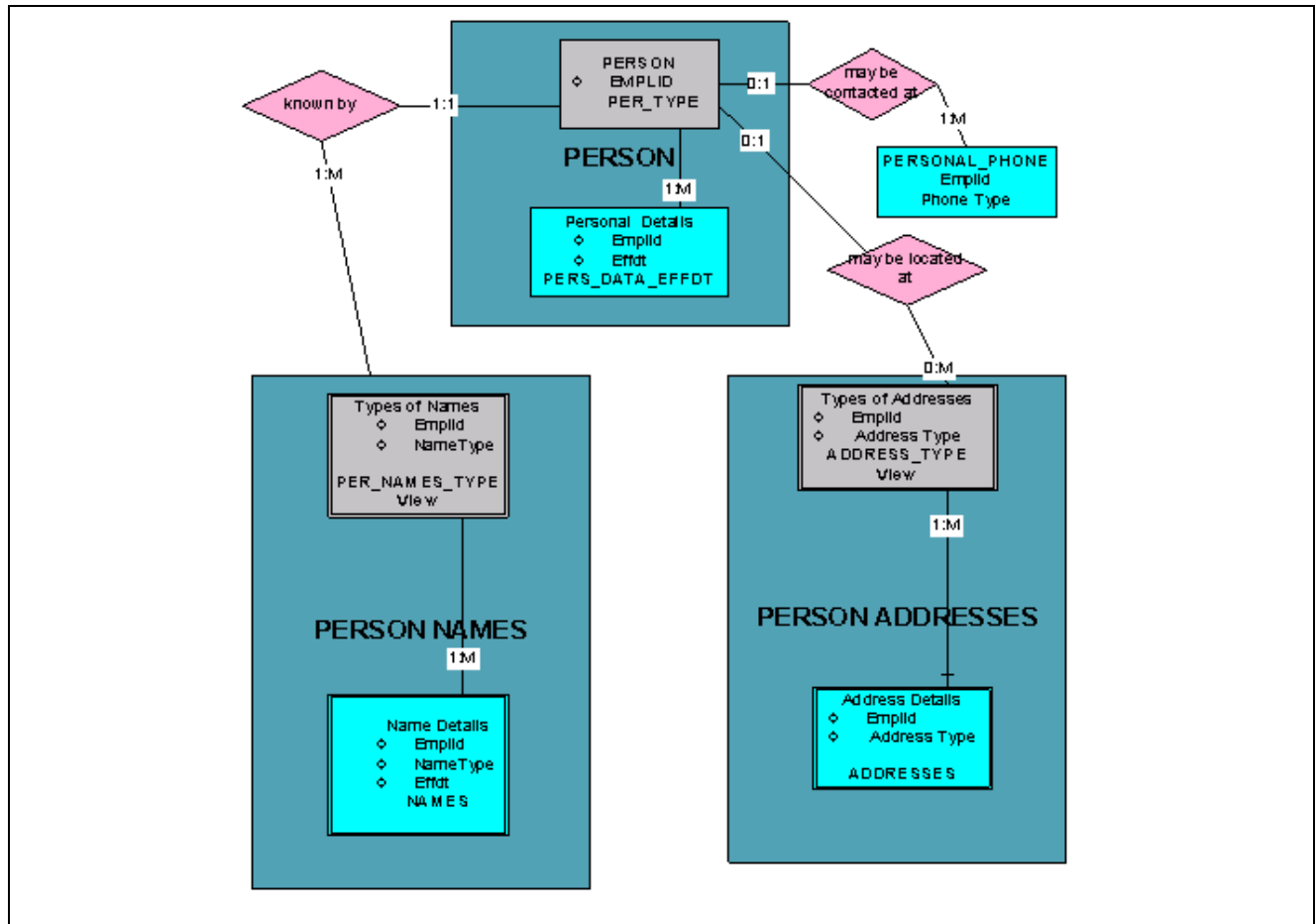
The following ERD describes the interrelationships of the PERSONAL\_DATA, PERS\_DATA\_EFFDT and other records where Addresses and Names are stored redundantly.



Partial Entity Relationship Diagram for Person Data, Names, and Addresses.

## Task P-1-2: Understanding PeopleSoft 8.8 Entity Relationships

The following Entity Relationship Diagram shows the relationship of Core Records in Release 8.8.



Entity Relationship Diagram of the Core Records in PeopleSoft 8.8

## Task P-1-3: Understanding PeopleSoft New Release Entity Relationships

This section discusses:

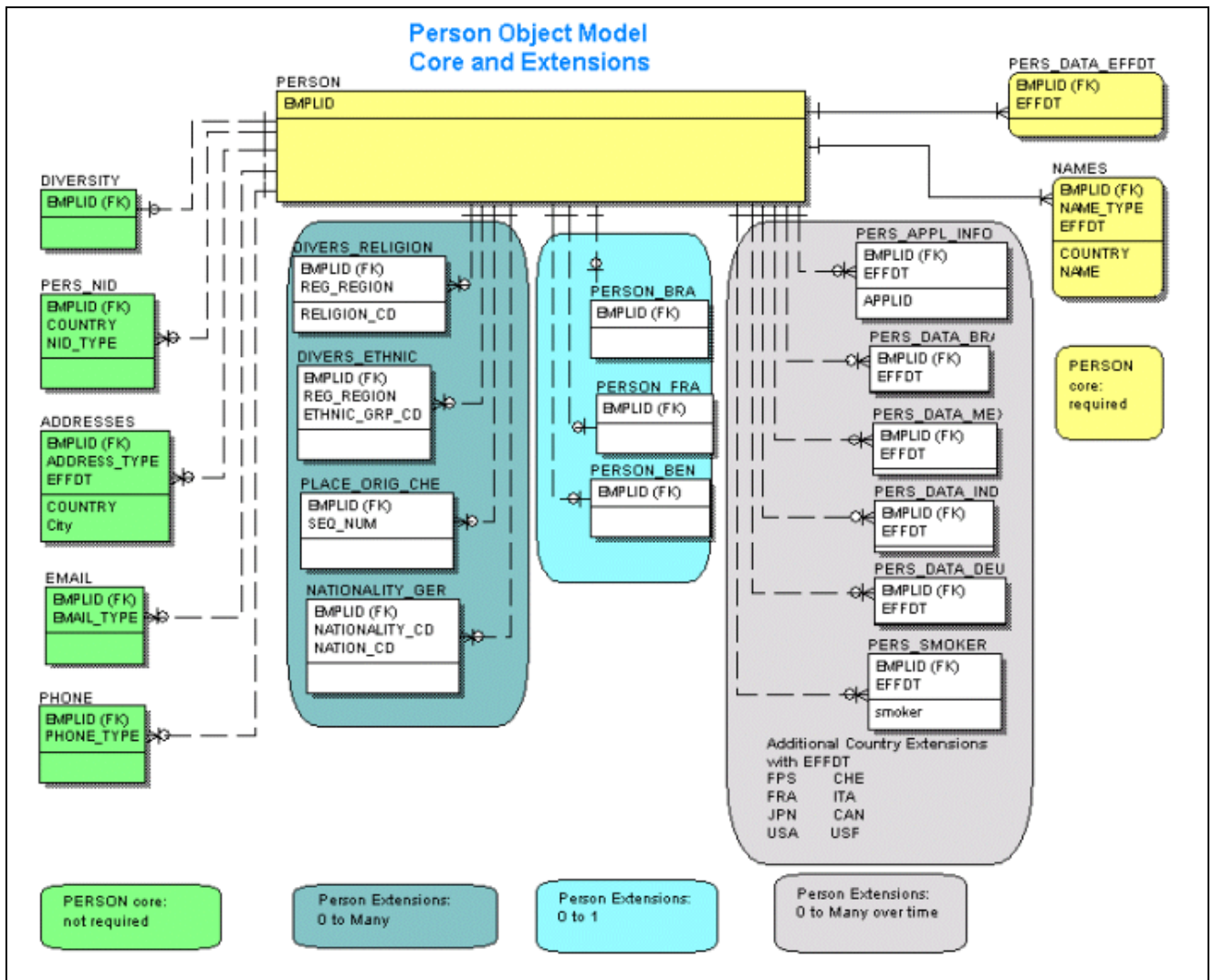
- Understanding PeopleSoft New Release Diagrams
- Person Object Model
- Person and Organization Entities
- Worker Entity
- What Happens to PERSONAL\_DATA
- What Happens to EMPLOYMENT

### Understanding PeopleSoft New Release Diagrams

The following Entity Relationship Diagrams (ERDs) provide a visual representation of some of the key relationships the Person Object has with other records in the new PeopleSoft release.

#### Person Object Model

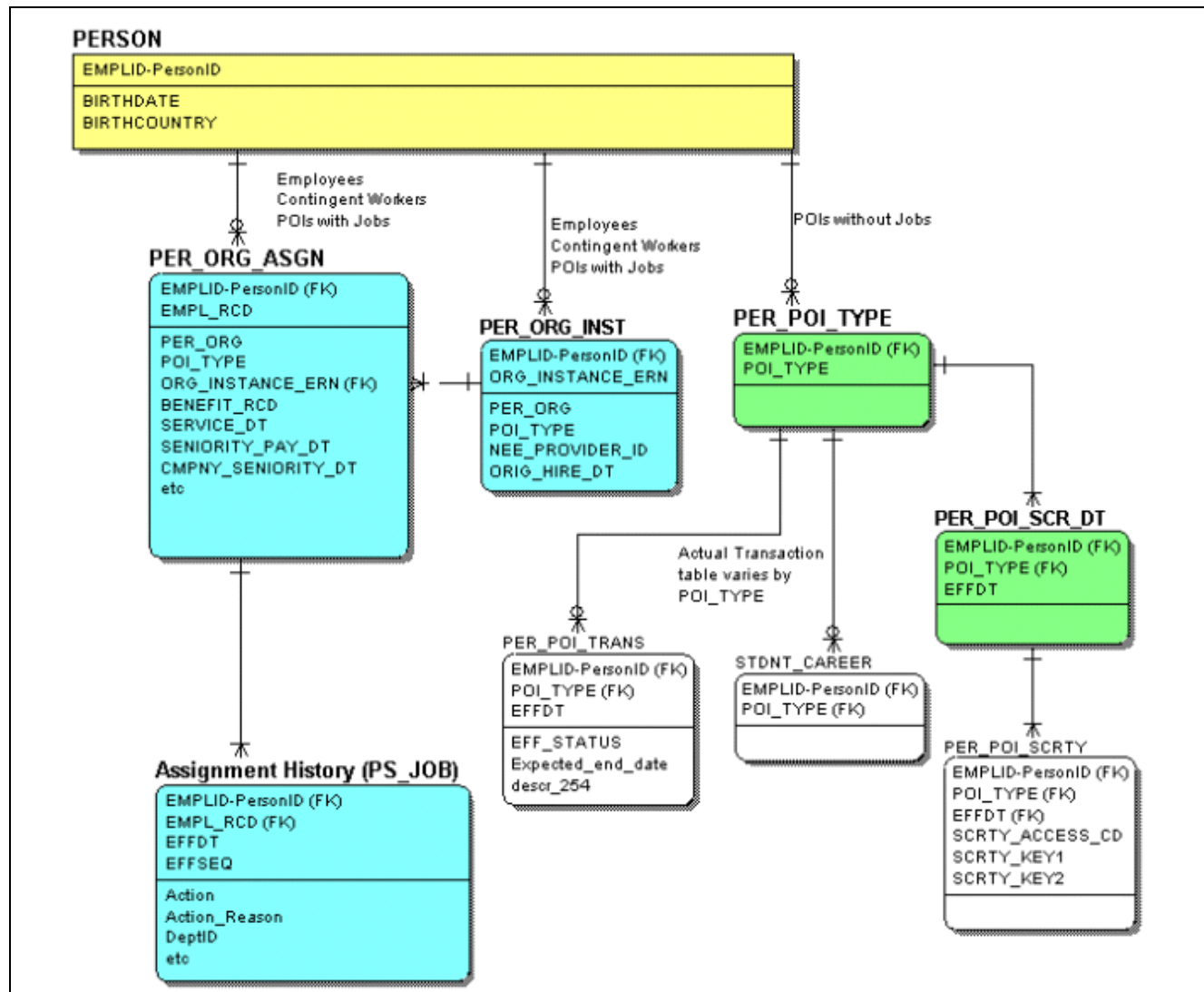
The following Entity Relationship Diagram illustrates the Person Object Model, showing core requirements and optional core records as well as the type and number of extensions.



Person Object Model ERD

## Person and Organization Entities

The following Entity Relationship Diagram shows the relationship between Person and Organizational Relationship entities.



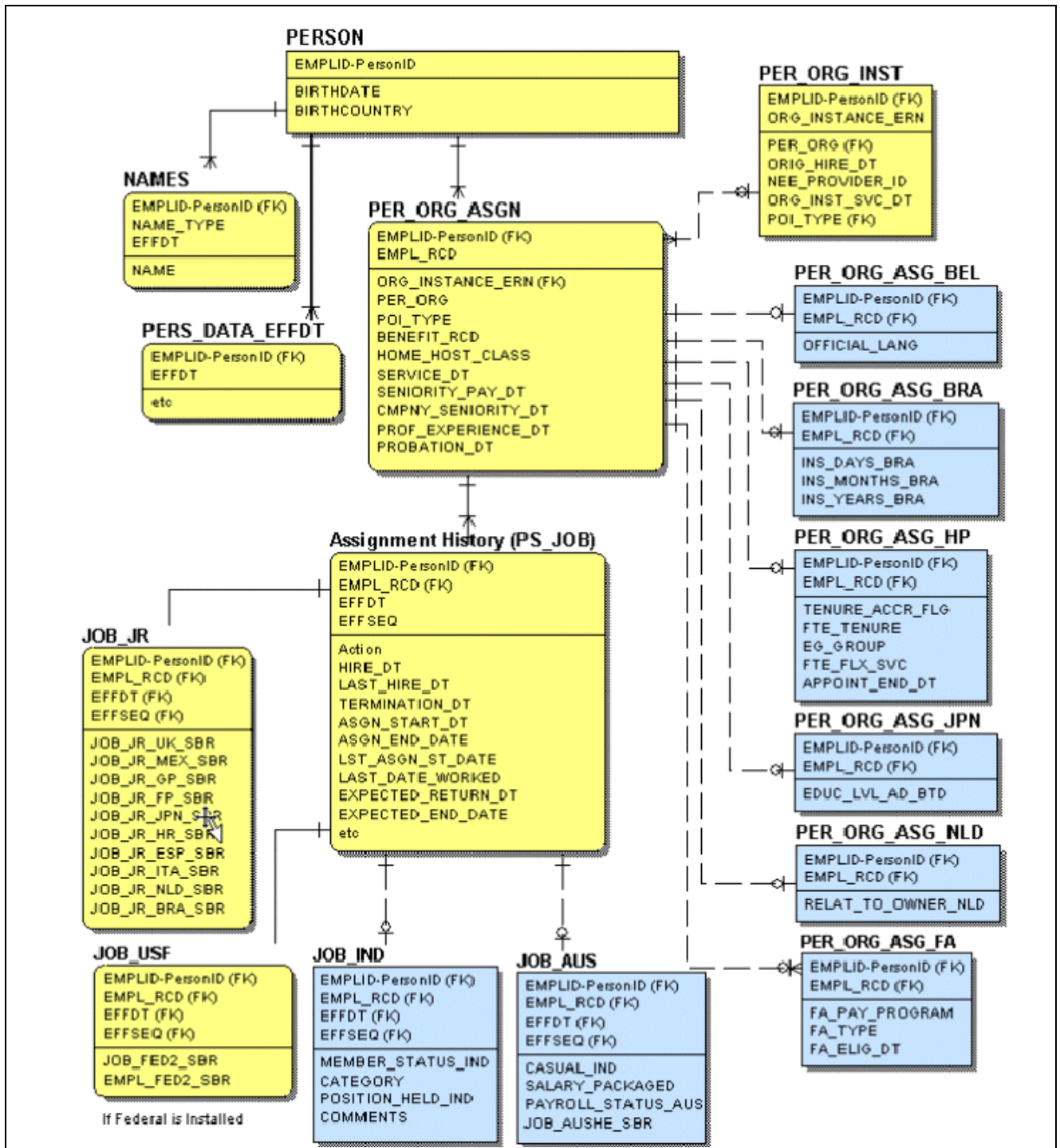
Person and Organizational Relationship ERD

While referencing the diagram above, keep in mind:

- While EMPLOYMENT is replaced by a new record (PER\_ORG\_ASGN), this record maintains the same key structure that EMPLOYMENT had.
- EMPLOYMENT remains as a View returning most of the same information.
- The key structure of JOB, COMPENSATION, and so on, will not change.
- A specific organizational relationship (Employee, Contingent Worker, POI) can be identified for each distinct EMPLID/EMPL\_RCD. This relationship will never change for that ENPL\_RCD. This is what allows us to keep the key structure of EMPLID, EMPL\_RCD throughout the rest of the system without having to add an additional key for the PER\_ORG.
- The PER\_ORG\_INST record is new, but it does represent a logical parent relationship that was already in the system. In order to keep the key structure of EMPLID and EMPL\_RCD intact for the rest of the system, we have created PER\_ORG\_INST as a logical parent not a physical parent. The relationship is captured using the common ORG\_INSTANCE\_ERN field on PER\_ORG\_INST and PER\_ORG\_ASGN.

## Worker Entity

The following illustrates the Worker entity relationships.



Worker ERD

## What Happens to PERSONAL\_DATA

While we are no longer using the PERSONAL\_DATA record as a core transaction table, we are keeping it for reporting and batch. PeopleSoft added processes to keep the data up to date in real-time with changes made to PERSON, PERS\_DATA\_EFFDT, NAMES, ADDRESSES, and the Person Extension records.

There is a component post change PeopleCode that will capture changes made in real-time. There is an Application Engine that can be run nightly to capture the data that has just become effective and also captures any changes that happened during the previous day. Please refer to the Application Fundamentals for HRMS PeopleBook for more information.

See *PeopleSoft Enterprise HRMS 9.0 Application Fundamentals PeopleBook*.

## What Happens to EMPLOYMENT

EMPLOYMENT is now a view that joins together the PER\_ORG\_ASGN, PER\_ORG\_INST, and current JOB row. It contains most of the same fields as the old EMPLOYMENT record and can still be used for reporting and online usage.

---

## Task P-2: Understanding Record and Table Changes

This section discusses:

- Understanding Record Changes
- Understanding Table Changes

### Task P-2-1: Understanding Record Changes

This section discusses:

- Record Changes from PeopleSoft 8.01
- Record Changes from PeopleSoft 8.8x
- Record Changes from PeopleSoft 8.01 and 8.8x

#### Record Changes from PeopleSoft 8.01

The following tables detail record changes from PeopleSoft 8.01:

PERSONAL_DATA	Fields deleted, added, moved	Conversion Rules
EMPLID	Moved to PERSON	
NAMEGBL_NPC_SBR	Delete	NAMES record created from PERS_DATA_EFFDT
ADDRESS_SBR	Delete	ADDRESSES record created from PERS_DATA_EFFDT
ADDR_OTR_SBR	Delete	ADDRESSES record created from PERS_DATA_EFFDT



<b>PERSONAL_DATA</b>	<b>Fields deleted, added, moved</b>	<b>Conversion Rules</b>
PHONE_SBR	Delete	PERSONAL_PHONE created from PERS_DATA_EFFDT
PER_STATUS	Delete	Replaced by PER_ORG_ASGN.PER_ORG
ORIG_HIRE_DT	Move to PER_ORG_INST	
SEX	Move to PERS_DATA_EFFDT	
AGE_STATUS	Move to PERS_DATA_DEU	
MAR_STATUS	Move to PERS_DATA_EFFDT	
BIRTHDATE	Move to PERSON	
BIRTHPLACE	Move to PERSON	
BIRTHCOUNTRY	Move to PERSON	
BIRTHSTATE	Move to PERSON	
DT_OF_DEATH	Move to PERSON	
HIGHEST_EDUC_LVL	Move to PERS_DATA_EFFDT	
FT_STUDENT	Move to PERS_DATA_EFFDT	
REFERRAL_SOURCE	Move to PERS_APPL_REF	
EMPL_REFERRAL_ID	Move to PERS_APPL_REF	
SPECIFIC_REFER_SRC	Move to PERS_APPL_REF	
HIGHLY_COMP_EMPL_P	Move to BN_PERSON	
HIGHLY_COMP_EMPL_C	Move to BN_PERSON	
RESUME_TEXT_FILE	Move to PERS_DATA_APPL	
QDRO_IND_YN	Deleted	
PER_TYPE	Move to PER_ORG_ASGN as POI_TYPE	
PERS_DTUSA_SBR	Move to PERS_DATA_USA	
PERS_DTCAN_SBR	Move to PERS_DATA_CAN	
PERS_DTCHE_SBR	Move to PERS_DATA_CHE	
PERS_DTGER_SBR	Move to PERS_DATA_DEU	

<b>PERSONAL_DATA</b>	<b>Fields deleted, added, moved</b>	<b>Conversion Rules</b>
PERS_DTFRA_SBR	Move to PERS_DATA_FRA	
PERS_DTJPN_SBR	Move to PERS_DATA_JPN	
PERS_DTUK_SBR	Move to PERS_DATA_UK	
PERS_DTITA_SBR	Move to PERS_DATA_ITA	
PERS_DTESP_SBR	Move to PERS_DATA_ESP	
LANG_CD	Move to PERS_DATA_EFFDT	
YEARS_OF_EXP	Move to PRIORWORK_SUM	
APPLID	Move to PERS_APPL_INFO	
APP_DT	Move to PERS_APPL_INFO	
ON_DM_EM_DOSAVENOW	Delete	
PERS_DTFED2_SBR	Move to PERS_DATA_USF	
ALTER_EMPLID	Move to PERS_DATA_EFFDT	
ADDRESS1_AC	Delete	See PERS_DATA_EFFDT
ADDRESS2_AC	Delete	See PERS_DATA_EFFDT
ADDRESS3_AC	Delete	See PERS_DATA_EFFDT
CITY_AC	Delete	See PERS_DATA_EFFDT

<b>PERS_DATA_EFFDT</b>	<b>Fields deleted, added, moved</b>	<b>Conversion Rules</b>
EMPLID		
EFFDT		
UPDATE_PERS_DATA	Delete	
ADDRESS_SBR	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION.ADDRESS_TYPE
ADDR_OTR_SBR	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION_ADDR_TYPE1
MAR_STATUS		
SMOKER	Move to PERS_SMOKER	
NAMEGBL_SBR	Move to NAMES	As NAME_TYPE = PRI

<b>PERS_DATA_EFFDT</b>	<b>Fields deleted, added, moved</b>	<b>Conversion Rules</b>
ADDRESS1_AC	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION.ADDRESS_TYPE
ADDRESS2_AC	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION.ADDRESS_TYPE
ADDRESS3_AC	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION.ADDRESS_TYPE
CITY_AC	Move to ADDRESSES	As ADDRESS_TYPE = INSTALLATION.ADDRESS_TYPE

### Record Changes from PeopleSoft 8.8x

The PeopleSoft 8.8x upgrade deals with creating the new structure that allows a single EMPLID to have more than one relationship to the organization. In addition, splitting the non-core data into components has been completed. Therefore, the following will not be done during your upgrade:

- The upgrade process will not attempt to merge EMPLIDs.
- The upgrade process will not assign the current EMPL\_RCDs to an EMPLID change.
- No EMPL\_RCDs will be deleted.

See *PeopleSoft Enterprise HRMS 9.0 Application Fundamentals PeopleBook*.

The following will occur during the upgrade:

- EMPL\_RCDs that were created with the HIR action will be created as separate Organizational Instances.
- EMPL\_RCDs that were created with the ADL action will be created as additional assignments and will be related to an appropriate Instance.
- EMPL\_RCDs that have Host assignments will be created as Instances.
- EMPL\_RCDs that are Host assignments will be created as Assignments related to the HOME assignment.
- Japanese main appointments will be created as Instances.
- Japanese additional assignments will be created as assignments connect to the Main appointment.

If you want EMPL\_RCDs that you created with the ADL action to be created as separate Instances, you should change that action to HIR in your source data prior to data conversion.

#### PERSON

<b>PERSON</b>	<b>Fields Deleted, Moved, Added</b>	<b>Conversion Rules</b>
EMPLID		
PER_STATUS	Replaced by PER_ORG_ASGN.PER_ORG	E = EMP N= See Chart 2 — Upgrade Conversion Chart below.
BIRTHDATE		
BIRTHPLACE		

PERSON	Fields Deleted, Moved, Added	Conversion Rules
BIRTHCOUNTRY		
BIRTHSTATE		
DT_OF_DEATH		
ORIG_HIRE_DT	Moved to PER_ORG_INST	Copied on each PER_ORG_INST row for a Person
HIGHLY_COMP_EMPL_C	Moved to BN_PERSON	Only create a row where there is data in either field. Since HIGHLY_COMP_EMPL_P had a defaulted specified of N this should be treated as a blank value.
HIGLY_COMP_EMPL_P	Moved to BN_PERSON	Same as above
LAST_CHILD_UPDTTM	New field	Null

*PERS\_DATA\_EFFDT*

PERS_DATA_EFFDT	Fields Deleted, Added, Moved	Conversion Rules
EMPLID		
EFFDT		
PER_TYPE	Replaced by POI_TYPE	See Chart 2 — Upgrade Conversion Chart below
MAR_STATUS		
MAR_STATUS_DT		
SEX		
AGE_STATUS	Moved to PERS_DATA_DEU	
HIGHEST_EDUC_LVL		
FT_STUDENT		
LANG_CD		
LASTUPDDTTM	New field	%datetime
LASTUPDOPRID	New field	PPLSOFT
YEARS_OF_EXP	Moved to PRIORWORK_SUM	
APPLID	Deleted — In PERS_APPL_INFO	If the APPLID/APP_DT is not PERS_APPL_INFO already, insert of row

<b>PERS_DATA_EFFDT</b>	<b>Fields Deleted, Added, Moved</b>	<b>Conversion Rules</b>
APP_DT	Delete — In PERS_APPL_INFO	See above
ALTER_EMPLID		
BILINGUALISM_CODE	PERS_DATA_CAN	
HEALTH_CARE_NBR	PERS_DATA_CAN	
HEALTH_CARE_STATE	PERS_DATA_CAN	
GUARDIAN_CHE	PERS_DATA_CHE	
MILIT_SITUATN_ESP	PERS_DATA_ESP	
SOC_SEC_AFF_DT	PERS_DATA_ESP	
PERS_DTFED2_SBR	PERS_DATA_USF	
ENTRY_DT_FRA	PERS_DATA_FRA	
MILIT_SITUATN_FRA	PERS_DATA_FRA	
CPAMID	PERS_DATA_FRA	
MILITARY_STAT_GER	PERS_DATA_DEU	
EXPCTD_MILITARY_DT	PERS_DATA_DEU	
HR_RESPONSIBLE_ID	PERS_DATA_DEU	
MILITARY_STAT_ITA	PERS_DATA_ITA	
MILATRY_TYPE_ITA	PERS_DATA_ITA	
MILIATARY_RANK_ITA	PERS_DATA_ITA	
MILITARY_END_ITA	PERS_DATA_ITA	
HONSEKI_JPN	PERS_DATA_JPN	
US_WORK_ELIGIBILITY	PERS_DATA_USA	
MILITARY_STATUS	PERS_DATA_USA	
CITIZEN_PROOF1	PERS_DATA_USA	
CITIZEN_PROOF2	PERS_DATA_USA	
SMOKER	PERS_SMOKER	
MEDICARE_ENTLD_DT	PERS_DATA_USA	
SMOKER_DT	PERS_SMOKER	

PERS_DATA_EFFDT	Fields Deleted, Added, Moved	Conversion Rules
FP_ACTION_2	PERS_DATA_FPS	
ACTION_REASON	PERS_DATA_FPS	
FP_ACTION_REQ	PERS_DATA_FPS	
FP_SUPDOC_REQ	PERS_DATA_FPS	
LAST_UPDATE_DATE	PERS_DATA_FPS	

Upgrade Conversion Chart					
PER_TYPE	Descr	PER_ORG	POI_TYPE	JOB	
C	Cobra Participant	POI	00001	Yes	COBRA Qualified Beneficiary
D	GP Dependent/Beneficiary	POI	00005	Yes	Global Payroll Payee
E	Leased Worker	CWR		Yes	Contingent Worker
H	Prehire	EMP		Yes	Employee
I	Independent Contractor	CWR		Yes	Contingent Worker
L	Time & Labor Contractor	CWR		Yes	Contingent Worker
M	Temporary Worker	CWR		Yes	Contingent Worker
N	Non HR Administered Employee	POI	00004	Yes	Stock - Non-HR Employee
O	Other Non-Employee	CWR		Yes	Contingent Worker
P	Pension Payee	POI	00002	Yes	Pension Payee
S	Stock Non-Employee	POI	00003	Yes	Stock - Board Member
T	Training Non-Employee	CWR		Yes	Contingent Worker
U	Succession Candidate Non-Emplo	CWR		Yes	Contingent Worker
Z	CS Student Refunds	POI	00006	Yes	CS Student Refund
Z	CS Student	POI	00009	No	CS Student

Chart 2 - Upgrade Conversion Chart for PER\_STATUS and PER\_TYPE

If the PER\_STATUS is *N* and there is no POI\_TYPE, the employee will be loaded as a Contingent Worker.

Since a Person can have different PERS\_DATA\_EFFDT.PER\_TYPES over time, it is possible for a person to have more than one row inserted into PER\_ORG\_ASGN.

## Record Changes from PeopleSoft 8.01 and 8.8x

The tables listed in this section are in both PeopleSoft 8.01 and 8.8x.

The upgrade converts EMPL\_RCDs that are in EMPLOYMENT to PER\_ORG\_ASGN and PER\_ORG\_INST. All rows in EMPLOYMENT will be moved to PER\_ORG\_ASGN. PER\_ORG\_INST will contain the relationship between EMPL\_RCDs in PER\_ORG\_ASGN. Some EMPL\_RCDs are considered Instances which actually creates a new employment instance. Others are additional assignments that the person may have over time but that should not be considered a "hire action." See the reference below for more information on the differences between Instances and Assignments.

See *PeopleSoft Enterprise HRMS 9.0 Application Fundamentals PeopleBook*.

EMPL\_RCDs that are Instances will also be in PER\_ORG\_INST. EMPL\_RCDs that are true additional assignments will not be in PER\_ORG\_INST, but will have a value in their PER\_ORG\_ASGN.ORG\_INSTANCE\_ERN that will connect them with an Instance.

The following will occur during the upgrade:

- EMPL\_RCDs that were created with the HIR action will be created as separate Organizational Instances.
- EMPL\_RCDs that were created with the ADL action will be created as additional assignments and will be related to an appropriate Instance.
- EMPL\_RCDs that have Host assignments will be created as Instances.
- EMPL\_RCDs that are Host assignments will be created as Assignments related to the HOME assignment.

- Japanese main appointments will be created as Instances.
- Japanese additional assignments will be created as assignments connect to the Main appointment.

If you want EMPL\_RCDs that you created with the ADL action to be created as separate Instances, you should change that action to *HIR* in your source data prior to data conversion.

### *NAMES*

<b>NAMES</b>	<b>Field Added</b>	<b>Conversion Rules</b>
EMPLID		
NAME_TYPE		
EFFDT		
COUNTRY_NM_FORMAT		Prompt Table is now NAME_FORMAT_TBL. Some values will need to be set to a default format value.
NAME_DISPLAY	New field	Casual formatted name which can be used on pages and reports. Formatting rules are based on the COUNTRY_NM_FORMAT.
NAME_FORMAL	New field	Formally formatted name which can be used for letters. Formatting rules are based on the COUNTRY_NM_FORMAT
LASTUPDDTTM	New field	%datetime
LASTUPDOPRID	New field	'PPLSOFT'

### *PERSONAL\_PHONE*

<b>PERSONAL_PHONE</b>	<b>Field Added</b>	<b>Conversion Rules</b>
EMPLID		
PHONE_TYPE		
COUNTRY_CODE		

<b>PERSONAL_PHONE</b>	<b>Field Added</b>	<b>Conversion Rules</b>
PHONE		
PREF_PHONE_FLAG	New field	<p>If you have SA, the type will be set based on the type selected in “Prepare Your Database,” Making Functional Decisions, Assigning Upgrade Default Values, Assigning Campus Community Upgrade Default Values first.</p> <p>If that type is not found or if you do not have SA, the order used in the upgrade is HOME, BUSN, MAIN or else the first found.</p>

*EMAIL\_ADDR*

<b>EMAIL_ADDRESS</b>	<b>Field Added</b>	<b>Conversion Rules</b>
EMPLID		
E_ADDR_TYPE		
EMAIL_ADDR		
PREF_EMAIL_FLAG	New field	Order is BUSN

*EMPLOYMENT*

<b>EMPLOYMENT</b>	<b>Replaced with a View based on new records</b>	<b>Conversion Rules</b>
EMPLID	Moved to PER_ORG_ASGN	
EMPL_RCD	Moved to PER_ORG_ASGN	
BENEFIT_RCD_NBR	Moved to PER_ORG_ASGN	
HOME_HOST_CLASS	Moved to PER_ORG_ASGN	
HIRE_DT	Moved to JOB	
REHIRE_DT	Moved to JOB as LST_HIRE_DT	
CMPNY_SENORITY_DT	Data copied to PER_ORG_ASGN.SEN_PAY_DT_OVR and PER_ORG_ASGN.CMPNY_SENIORITY_DT	
SERVICE_DT	Moved to PER_ORG_ASGN	
PROF_EXERIENCE_DT	Moved to PER_ORG_ASGN	



EMPLOYMENT	Replaced with a View based on new records	Conversion Rules
LAST_VERIFICATN_DT	Moved to PER_ORG_ASGN	
EXPECTED_RETURN_DT	Moved to JOB	
TERMINATION_DT	Moved to JOB	
LAST_DATE_WORKED	Moved to JOB	
LAST_INCREASE_DT	Moved to PER_ORG_ASGN	
OWN_5PERCENT_CO	Moved to PER_ORG_ASGN	
BUSINESS_TITLE	Moved to PER_ORG_ASGN	
REPORTS_TO	Delete (was moved to JOB in PeopleSoft 8.0)	
SUPERVISOR_ID	Delete (was moved to JOB in PeopleSoft 8.0)	
PROBATION_DT	Moved to PER_ORG_ASGN	
COUNTRY_CODE	Moved to PER_ORG_ASGN	
PHONE	Moved to PER_ORG_ASGN as POSITION_PHONE	
TIME_RPT_LOC	Deleted — not used	
JOB_REPORTING	Deleted — not used	
DED_TAKEN	Moved to DED_NA	
DED_SUBSET_ID	Moved to DED_NA	
GVT_SCD_RETIRE	Moved to JOB_USF	
GVT_SCD_TSP	Moved to JOB_USF	
GVT_SCD_LEO	Moved to JOB_USF	
GVT_SCD_SEVPAY	Moved to JOB_USF	
GVT_SECPAY_PRV_WRKS	Moved to JOB_USF	
GVT_MAND_RET_DAT	Moved to JOB_USF	
GVT_WGI_STATUS	Moved to JOB_USF	
GVT_INTRM_DAYS_WGI	Moved to JOB_USF	
GVT_NONPAY_NOA	Moved to JOB_USF	

<b>EMPLOYMENT</b>	<b>Replaced with a View based on new records</b>	<b>Conversion Rules</b>
GVT_NONPAY_HRS_WGI	Moved to JOB_USF	
GVT_NONPAY_HRS_SCD	Moved to JOB_USF	
GVT_NONPAY_HRS_TNR	Moved to JOB_USF	
GVT_NONPAY_HRS_PRB	Moved to JOB_USF	
GVT_TEMP_PRO_EXPIR	Moved to JOB_USF	
GVT_TEMP_PSN_EXPIR	Moved to JOB_USF	
GVT_DETAIL_EXPIRES	Moved to JOB_USF	
GVT_SABBATIC_EXPIR	Moved to JOB_USF	
GVT_RTND_GRADE_BEG	Moved to JOB_USF	
GVT_RTND_GRADE_EXP	Moved to JOB_USF	
GVT_NOA_CODE	Moved to JOB_USF	
GVT_CURR_APT_AUTH1	Moved to JOB_USF	
GVT_CURR_APT_AUTH2	Moved to JOB_USF	
GVT_APPT_EXPIR_DT	Moved to JOB_USF	
GVT_CNV_BEGIN_DATE	Moved to JOB_USF	
GVT_CAREER_CNV_DUE	Moved to JOB_USF	
GVT_CAREER_COND_DT	Moved to JOB_USF	
GVT_APPT_LIMIT_HRS	Moved to JOB_USF	
GVT_APPT_LIMIT_DYS	Moved to JOB_USF	
GVT_APPT_LIMIT_AMT	Moved to JOB_USF	
GVT_SUPV_PROB_DT	Moved to JOB_USF	
GVT_SES_PROB_DT	Moved to JOB_USF	
GVT_SEC_CLR_STATUS	Moved to JOB_USF	
GVT_CLRNCE_STAT_DT	Moved to JOB_USF	
GVT_ERN_PGM_PERM	Moved to JOB_USF	
GVT_OCC_SERS_PERM	Moved to JOB_USF	

<b>EMPLOYMENT</b>	<b>Replaced with a View based on new records</b>	<b>Conversion Rules</b>
GVT_GRADE_PERM	Moved to JOB_USF	
GVT_COMP_AREA_PERM	Moved to JOB_USF	
GVT_COMPA_LVL_PERM	Moved to JOB_USF	
GVT_CHANGE_FLAG	Moved to JOB_USF	
GVT_SPEP	Moved to JOB_USF	
GVT_WGI_DUE_DATE	Moved to JOB_USF	
GVT_DT_LEI	Moved to JOB_USF	
GVT_FIN_DISCLOSURE	Moved to JOB_USF	
GVT_FIN_DISCL_DATE	Moved to JOB_USF	
GVT_TENURE	Moved to JOB_USF	
GVT_DETL_BARG_UNIT	Moved to JOB_USF	
GVT_DETL_UNION_CD	Moved to JOB_USF	
NEXT_REVIEW_DATE	Moved to JOB_USF	
GVT_WELFARE_WK_CD	Moved to JOB_USF	
TENURE_ACCR_FLG	Moved to PER_ORG_ASG_HP	
FTE_TENURE	Moved to PER_ORG_ASG_HP	
EG_GROUP	Moved to PER_ORG_ASG_HP	
FTE_FLX_SRVC	Moved to PER_ORG_ASG_HP	
CONTRACT_LENGTH	Moved to PER_ORG_ASG_HP	
APPOINT_END_DT	Moved to PER_ORG_ASG_HP	
NEE_PROVIDER_ID	Moved to PER_ORG_INST	
FA_PAY_PROGRAM	Moved to PER_ORG_ASG_FA	
FA_TYPE	Moved to PER_ORG_ASG_FA	
FA_ELIG_DT	Moved to PER_ORG_ASG_FA	
EDUC_LVVL_AD_BT	Moved to PER_ORG_ASG_JPN	
OFFICIAL_LANG	Moved to PER_ORG_ASG_BEL	

<b>EMPLOYMENT</b>	<b>Replaced with a View based on new records</b>	<b>Conversion Rules</b>
RELAT_TO_OWNER_NLD	Moved to PER_ORG_ASG_NLD	
INS_DAYS_BRA	Moved to PER_ORG_ASG_BRA	
INS_MOTHS_BRA	Moved to PER_ORG_ASG_BRA	
INS_YEARS_BRA	Moved to PER_ORG_ASG_BRA	

*JOB*

<b>JOB</b>	<b>Fields Added or Deleted</b>	<b>Conversion Rules</b>
EMPLID		
EMPL_RCD		
EFFDT		
EFFSEQ		
PER_ORG	New	
HR_STATUS	New	
GVT_EFFDT	Moved to JOB_USF	
GVT_EFFDT_PROPOSED	Moved to JOB_USF	
GVT_TRANS_NBR	Moved to JOB_USF	
GVT_TRANS_NBR_SEQ	Moved to JOB_USF	
GVT_WIP_STATUS	Moved to JOB_USF	
GVT_STATUS_TYPE	Moved to JOB_USF	
GVT_NOA_CODE	Moved to JOB_USF	
GVT_LEG_AUTH_1	Moved to JOB_USF	
GVT_PAR_AUTH_D1	Moved to JOB_USF	
GVT_PAR_AUTH_D1_2	Moved to JOB_USF	
GVT_LEG_AUTH_2	Moved to JOB_USF	
GVT_PAR_AUTH_D2	Moved to JOB_USF	
GVT_PAR_AUTH_D2_2	Moved to JOB_USF	
GVT_PAR_NTE_DATE	Moved to JOB_USF	

<b>JOB</b>	<b>Fields Added or Deleted</b>	<b>Conversion Rules</b>
GVT_WORK_SCHED	Moved to JOB_USF	
GVT_SUB_AGENCY	Moved to JOB_USF	
GVT_ELIG_FEHB	Moved to JOB_USF	
GVT_FEHB_DT	Moved to JOB_USF	
GVT_PAY_RATE_DETER	Moved to JOB_USF	
GVT_STEP	Moved to JOB_USF	
GVT_RTND_PAY_PLAN	Moved to JOB_USF	
GVT_RTND_SAL_PLAN	Moved to JOB_USF	
GVT_RTND_GRADE	Moved to JOB_USF	
GVT_RTND_STEP	Moved to JOB_USF	
GVT_RTND_GVT_STEP	Moved to JOB_USF	
GVT_PAY_BASIS	Moved to JOB_USF	
GVT_COMPRATE	Moved to JOB_USF	
GVT_LOCALITY_ADJ	Moved to JOB_USF	
GVT_BIWEEKLY_RT	Moved to JOB_USF	
GVT_DAILY_RT	Moved to JOB_USF	
GVT_HRLY_RT_NO_LOC	Moved to JOB_USF	
GVT_DLY_RT_NO_LOC	Moved to JOB_USF	
GVT_BW_RT_NO_LOC	Moved to JOB_USF	
GVT_MNLY_RT_NO_LOC	Moved to JOB_USF	
GVT_ANNL_RT_NO_LOC	Moved to JOB_USF	
GVT_XFER_FROM_AGCY	Moved to JOB_USF	
GVT_XFER_TO_AGCY	Moved to JOB_USF	
GVT_RETIRE_PLAN	Moved to JOB_USF	
GVT_ANN_IND	Moved to JOB_USF	
GVT_Fegli	Moved to JOB_USF	

<b>JOB</b>	<b>Fields Added or Deleted</b>	<b>Conversion Rules</b>
GVT_FEGLI_LIVING	Moved to JOB_USF	
GVT_LIVING_AMT	Moved to JOB_USF	
GVT_ANNUITY_OFFESET	Moved to JOB_USF	
GVT_CSRS_FROZN_SVC	Moved to JOB_USF	
GVT_GREV_RET_COVRG	Moved to JOB_USF	
GVT_FERS_COVERAGE	Moved to JOB_USF	
GVT_TYPE_OF_APPT	Moved to JOB_USF	
GVT_POI	Moved to JOB_USF	
GVT_POSN_OCCUPIED	Moved to JOB_USF	
GVT_CONT_EMPLID	Moved to JOB_USF	
GVT_ROUTE_NEXT	Moved to JOB_USF	
GVT_CHANGE_FLAG	Moved to JOB_USF	
GVT_TSP_UPD_IND	Moved to JOB_USF	
GVT_PI_UPD_IND	Moved to JOB_USF	
GVT_SF52_NBR	Moved to JOB_USF	
GVT_S113G_CEILING	Moved to JOB_USF	
GVT_LEO_POSITION	Moved to JOB_USF	
GVT_ANNUIT_COM_DT	Moved to JOB_USF	
GVT_BASIC_LIFE_RED	Moved to JOB_USF	
GVT_DED_PROPT_DT	Moved to JOB_USF	
GVT_FEGLI_BASC_PCT	Moved to JOB_USF	
GVT_FEGLI_OPT_PCT	Moved to JOB_USF	
GVT_FEHB_PCT	Moved to JOB_USF	
GVT_RETRO_FLAG	Moved to JOB_USF	
GVT_RETRO_DED_FLAG	Moved to JOB_USF	
GVT_RETRO_JOB_FLAG	Moved to JOB_USF	
GVT_RETRO_BSE_FLAG	Moved to JOB_USF	

JOB	Fields Added or Deleted	Conversion Rules
GVT_OTH_PAY_CHG	Moved to JOB_USF	
GVT_DETL_POSN_NBR	Moved to JOB_USF	
ANNL_BEN_BASE_OVRD	Moved to JOB_USF	
BENEFIT_PROGRAM	Moved to JOB_USF	
UPDATE_PAYROLL	Moved to JOB_USF	
GVT_PAY_PLAN	Moved to JOB_USF	
GVT_PAY_FLAG	Moved to JOB_USF	
GVT_NID_CHANGE	Moved to JOB_USF	
WPP_STOP_FLAG	New	
SUPV_LVL_ID	Moved from JOB_JR	
SETID_SUPV_LVL	Moved from JOB_JR	
ABSENCE_SYSTEM_FLG	New	
POI_TYPE	New	
AUTO_END_FLG	Moved from JOB_JR	
HIRE_DT	Moved from EMPLOYMENT	
LAST_HIRE_DT	Moved from EMPLOYMENT	
TERMINATION_DT	Moved from EMPLOYMENT	
ASGN_START_DT	New	
LST_ASSGN_START_DT	New	
ASGN_END_DT	New	
LDW_OVR	New	
LAST_DATE_WORKED	Moved from EMPLOYMENT	
EXPECTED_RETURN_DT	Moved from EMPLOYMENT	
EXPECTED_END_DATE	Moved from JOB_JR	
LASTUPDDTM	New	
LASTUPDOPRID	New	

*JOB\_JR*

<b>JOB_JR</b>	<b>Field Deleted, Added or Moved</b>	<b>Conversion Rules</b>
EMPLID		
EMPL_RCD		
EFFDT		
EFFSEQ		
CASUAL_IND	Moved to JOB_AUS	
SALARY_PACKAGED	Moved to JOB_AUS	
PAYROLL_STATE_AUS	Moved to JOB_AUS	
CLASSN_CURRENT_AUS	Moved to JOB_AUS	
WORK_SECTOR_AUS	Moved to JOB_AUS	
FUNCTION_AUS	Moved to JOB_AUS	
ANN_CNTRACT_HRS_AUS	Moved to JOB_AUS	
TEACH_WEEKS_AUS	Moved to JOB_AUS	
CASUAL_TYPE_AUS	Moved to JOB_AUS	
TERM_TYPE_AUS	Moved to JOB_AUS	
TERM_LTD_AUS	Moved to JOB_AUS	
SUPV_LVL_ID	Moved to JOB	
EXPECTED_END_DATE	Moved to JOB	
AUTO_END_FLAG	Moved to JOB	

Because of the previous model, it is assumed that all EMPLOYMENT records go with the PER\_ORG based on the PER\_STATUS. All EMPLOYMENT records belong to *one* PERSON\_ORG. The one exception is for Pension Payees. If the Person has a PER\_STATUS of *E* and the PER\_TYPE of *P* then their last EMPL\_RCD will be for their Pension data and should get a PER\_ORG of POI.

See “Complete Database Changes,” task Converting Retirees with Pay for information about converting retirees with pay.

## Task P-2-2: Understanding Table Changes

This section discusses:

- PERSON Record
- PERS\_DATA\_EFFDT Record



- NAMES
- ADDRESSES
- DEPENDENT\_BENEF
- EMERGENCY\_CNTCT
- PAYROLL\_DATA
- ABS\_HIST\_DET
- ADDRESS\_TYPE
- PERS\_NAMES\_TYPE
- New Views
- PERSON\_NAME
- PERSON\_ADDRESS
- PERSONAL\_DT\_FST

## **PERSON Record**

PERSON is a new Record that serves as the parent to PERS\_DATA\_EFFDT, EMPLOYMENT, and all other records that previously used PERSONAL\_DATA as the parent.

The upgrade scripts will load this record with the information from PERSONAL\_DATA.

## **PERS\_DATA\_EFFDT Record**

The upgrade scripts will ensure that all PERS\_DATA\_EFFDT rows are updated with the data from PERSON for the fields moved from PERSONAL\_DATA to PERS\_DATA\_EFFDT.

## **NAMES**

This is the only source of Name data for a person in the record, PERSON.

The upgrade scripts will ensure that a row is created in NAMES for any PERS\_DATA\_EFFDT row that had a name change for NAME\_TYPE = 'PRI'.

The fields NAME\_DISPLAY and NAME\_FORMAL are new in this PeopleSoft release. To populate them, you must run the Application Engine process NAME\_DISPLAY. This process is called during data conversion. If the fields are not populated after data conversion you can run the Refresh Name Display Values process at any time.

To run the Refresh Name Display Values process:

Select Set Up HRMS, System Administration, Database Processes, Refresh Name Display Values.

To update all name formats check Update All Name Formats. Otherwise, select a Name format Type. Be sure that the following check boxes are selected:

- Update NAMES record
- Update Dependants
- Update HS Non-Employees
- Update French Public Sector
- Update US Federal Names

## ADDRESSES

This is the only source of Address data for a person in the record PERSON.

The upgrade scripts will ensure that a row is created in ADDRESSES for any PERS\_DATA\_EFFDT row that had an address change for the ADDRESS\_TYPE = INSTALLATION.ADDRESS\_TYPE if you are upgrading from 8.01.

The upgrade scripts will ensure that a row is created in ADDRESSES for any PERS\_DATA\_EFFDT row that had another address change for the ADDRESS\_TYPE = INSTALLATION.ADDR\_TYPE1 if you are upgrading from 8.01.

## DEPENDENT\_BENEF

If the Address is the same as the employee's then the address data is not physically stored on the record, only the Address\_type is. If the Address is not the same as the employee's, then the dependent's address is directly stored on the record.

If the Phone is the same as the employees, then the phone data is not physically stored on the record, only the PHONE\_TYPE is. If the Phone is not the same as the employee's, then the dependent's phone is directly stored on the record. There is now the ability to have the address be the same as the employee's while the Phone is different.

The Upgrade scripts will enter an Address\_type of HOME and a Phone\_Type of MAIN if the Same Address as Employee is Yes. The address and phone fields will then be cleared. No change will happen if the Same Address as Employee is No.

## EMERGENCY\_CNTCT

If the address is the same as the employee's, then the address data is not physically stored on the record, only the Address\_type is stored. If the address is not the same as the employee's, then the contact's address is directly stored on the record.

If the phone is the same as the employee's, then the phone data is not physically stored on the record, only the PHONE\_TYPE is stored. If the phone is not the same as the employee's, then the contact's phone is directly stored on the record. It is now possible for the address to be the same as the employee's while the phone is different.

If you are upgrading from Peoplesoft 8.01, when Same Address as Employee option is set to *Yes*:

- The Upgrade scripts will enter the Address\_type from PS\_INSTALLATION table.
- The Phone\_Type is set to the same as the preferred phone from the PS\_PERSONAL\_PHONE.
- The address and phone fields will be cleared.

No change will happen if the Same Address as Employee option is set to *No*.

## PAYROLL\_DATA

The Address fields were removed from PAYROLL\_DATA. Any address used to send the paycheck or advice to must exist in ADDRESSES. However, the PAYROLL\_DATA component allows the user to select the HOME or MAILING address or to enter a different CHECK address.

The Upgrade scripts will create an ADDRESSES record with the data from PAYROLL\_DATA with the appropriate ADDRESS\_TYPE if this data does not exist in ADDRESSES.

## **ABS\_HIST\_DET**

If the Address is the same as the employee's then the address data is not physically stored on the record, only the ADDRESS\_TYPE is. If the Address is not the same as the employee's, then the address is directly stored on the record.

The Upgrade scripts will enter an Address\_type of HOME and a Phone\_Type of MAIN if the Same Address as Employee is Yes. The address and phone fields will then be cleared. No change will happen if the Same Address as Employee is No.

## **ADDRESS\_TYPE**

This has been replaced by a table built from ADDRESSES.

## **PERS\_NAMES\_TYPE**

This has been replaced by a table built from NAMES.

## **New Views**

The following lists the new views that have been created.

### **PERSON\_NAME**

This view returns a Person's current Primary Name. One row per PERSON will always exist in this view.

### **PERSON\_ADDRESS**

This view returns a Person's current Address of each ADDRESS\_TYPE. One row per Person per Address\_type will exist in this view. Since Addresses are not required, this view will not return all people in PERSON.

### **PERSONAL\_DT\_FST**

This view is a combination of PERSON, current primary name from NAMES, and the current row from PERS\_DATA\_EFFDT. One row will exist for each person in PERSON.



## APPENDIX Q

# Understanding Service Indicator Changes

This appendix discusses:

- Reviewing Service Indicator Code Upgrade Defaults
- Understanding Basis of Impact
- Analyzing Current Service Impacts and Associations
- Understanding Service Indicator Audits
- Understanding Service Indicator Display Security

---

### Task Q-1: Reviewing Service Indicator Code Upgrade Defaults

You should review the changes that have been made to Service Indicator code definitions in PeopleSoft 9. In addition to new functionality that allows you to select whether to use Service Indicator codes with persons or organizations (see “Prepare Your Database,” Making Functional Decisions, Upgrading Defaults for Service Indicators), Service Indicator code definitions include three other new fields, which do not require you to define upgrade defaults:

- **Default Start Date:** If this option is selected for a Service Indicator code, when the code is assigned, it will use the default current system date as the start date.

The Default Start Date check box is cleared during the upgrade process to avoid unnecessary user intervention.

- **Default Start Term:** If this option is selected for a Service Indicator code, when the code is assigned, it will use the default value *0000* (Begin Term – Srvc Indicatr Use) as the start term.

The Default Start Term check box is selected during the upgrade process in order to most closely match previous functionality while avoiding unnecessary user interaction.

- **Display in Self Service:** If this option is selected for a Service Indicator code, the code, when assigned, will be visible to a user in Self Service.

Previous functionality allowed only the display of negative Service Indicators (Holds) in Self Service. During the upgrade process, this check box option will be selected for negative Service Indicators and cleared for positive Service Indicators, in order to continue previous functionality without user intervention.

You should analyze the default values for all of these fields on a code-by-code basis after upgrading and make any changes, as needed.

---

## Task Q-2: Understanding Basis of Impact

The most significant change to Service Indicators is that Service Impacts now include the attribute Basis of Impact. This provides a consistent logic across modules and products regarding the validity period of an impact; that is, when or whether an impact is currently in effect for an assignee.

In previous releases, a Service Indicator assignment included an Active Term and Active Date; however, because of different functional requirements, various modules within Campus Solutions defined those fields differently. For example, in Financial Aid, Active Term was the only term for which a Service Indicator was in effect; for Enrollment, Active Term was the first term for which a Service Indicator was in effect, and the Service Indicator was in effect until removed.

In Campus Solutions 9, Active Term and Active Date have been changed to Start Term and Start Date, respectively, and the new fields End Term and End Date have been added to allow all modules to define these fields in a consistent manner. In order for the software to determine whether an impact should consider the Date period (Start Date to End Date), Term period (Start Term to End Term), or both, an impact now includes Basis of Impact as an attribute.

Due to this change, it is possible that Service Indicator definitions will exist in your current database that will have associated impacts, each of which in previous releases interpreted Active Term in a different manner. Because Start Term and End Term will now always be interpreted consistently, the functional implications of the Active Term value will need to be considered. If a single Service Indicator definition includes “mixed” impacts (that is, some impacts that define Active Term as the only term, and some that define it as the first term), you may need to split these codes into new definitions to preserve the functional intent of the original assignment.

For example, suppose that Service Indicator Code SF9 is created, and it has Service Impacts CENR (block all enrollments), FAD (block Financial Aid disbursement), and CHECK (block check cashing privileges) associated with it. SF9 is assigned to Bob Smith, EMPLID 1100, with an Active Term of 0560 and an Active Date of 09/01/2006. The Enrollment engine runs, processing enrollments for Term 0570, finds this assigned Service Indicator for Bob Smith, and blocks his enrollment, since 0570 is equal to or greater than 0560. (The enrollment process interpreted Active Term as the first term for which the Service Indicator was in effect). The Financial Aid process runs, finds the impact of FAD, but processes the disbursement, because 0570 > 0560. (The Financial Aid process interpreted Active Term as the *only* term for which the Service Indicator was in effect.)

When you upgrade to PeopleSoft 9, Active Term will change to Start Term, so a direct upgrade will provide the following results: Bob Smith, SF9, Start term = 0560. Now when the Enrollment engine runs, it uses a Start Term value of 0560 and blocks the enrollment successfully. But when disbursements are processed, it *also* blocks the disbursement, because Start Term now always means the first term for which the Service Indicator and its impacts are in effect.

However, if we conversely automatically set an End Term of 0560 (which in PeopleSoft 9 is the functional equivalent of a single term validity period), disbursement will be properly processed and released for term 0570, but enrollment will incorrectly fail to block enrollment processing because the term validity has expired. If you upgrade these two impacts to the same Service Indicator, you will in the process redefine the effect on any assignments of that Service Indicator.

## Task Q-3: Analyzing Current Service Impacts and Associations

The first step that you should take in upgrading Service Indicators is to analyze your existing service impacts and their associations with existing Service Indicator codes. With your institutional knowledge of how those impacts and codes are being used, you will need to determine the following:

- The basis of impact for each Service Impact.

That is, in which of the following ways is the Service Impact being used:

- In a date-based context (for example, in Student Financials, Transcripts, most Contributor Relations impacts, and many Campus Community-oriented impacts, such as a library fine).
- In a term-based context (for example, Financial Aid, or possibly some Campus Community-oriented impacts).
- In a context that is both date-based and term-based. (For example, Enrollments uses both the date, to determine the initial validity period, and the term, to determine whether the term being processed falls within the term validity period.)
- The current association of impacts to Service Indicator codes.

If a code has impacts that would cause a direct mapping to corrupt the original functional intent of the assignment, you will need to “split” that code definition into two or more new code definitions.

A basic SQR report, PUCCR10.SQR, is provided to show you the current Service Impact and code associations. You can use this report to assist you in understanding your current definitions and how you may need to change them, based on the information above.

To upgrade Service Indicator code and impact associations, you will need to decide what values to use for the End Term Equals, “Convert 9999 to,” and Map fields on the Upgrade Service Indicator Code/Impact page:

- End Term Equals value

Because End Term is a new field in PeopleSoft 9, and the functionality in previous releases allowed for different interpretations of the Active Term field, for each code you must select a value for End Term Equals to which you want to upgrade assigned Service Indicators with that code. You should select the value that preserves the functional intent of the assignment.

For example, previously the Service Indicator Code FA1 (Financial Aid 1) interpreted Active Term as the only term for which the assignment was in effect. In PeopleSoft 9, this is equivalent to a Start Term and End Term with the same value (a Start Term of 0560, and an End Term of 0560); that is, any term-based impacts on this Service Indicator Code are interpreted as taking effect when Term 0560 begins, and no longer being in effect when Term 0560 has ended. Therefore, for the code FA1, for the End Term Equals value, you would select the Start Term option.

In another example, the Service Indicator Code REG has several impacts, including CENR. CENR interprets Active Term to mean the first term for which it is in effect and all subsequent terms. In PeopleSoft 9, this is equivalent to a Start Term of 0560 and an End Term set to Blank. There is no defined End Term, and the impact is in effect for all terms subsequent to and including the Start Term. Therefore, for the code REG, for the End Term Equals value, you would select the Blank option.

- Convert 9999 to

Similar to End Term, in previous releases, an Active Term value of 9999 was interpreted by different modules to mean different things. In Financial Aid, an Active Term of 9999 was equivalent to “all terms;”

that is, the Service Indicator was in effect indefinitely. For Enrollment, an Active Term of 9999 meant “no terms;” that is, the Service Indicator, though assigned, would never actually create an impact.

In PeopleSoft 9, the addition of the End Term field removes much of the original functional need for a 9999 value. In PeopleSoft 9, the value 9999 becomes a term infinitely in the future; that is, a term that will never be reached. A Start Term value of 9999 will never take effect (it will not start until some unreachable point in the future). An End Term value of 9999 will be in effect indefinitely and is equivalent, therefore, to an End Term set to Blank. To preserve flexibility of usage, 9999 remains a valid value for Start Term and End Term, but you should be aware of how it is interpreted in PeopleSoft 9.

As with End Term, therefore, for each Service Indicator code you will need to decide how to upgrade the value 9999 for Start Term so that the original functional intent is preserved. For example, Service Indicator Code FA1 interpreted an Active Term of 9999 to mean “all terms.” The functional equivalent of this in PeopleSoft 9 is a Start Term of 0000 (start term is infinitely in the past, and no End Term is defined). Therefore, for the “Convert 9999 to” field, you would select the value 0000.

---

**Note.** The “Convert 9999 to” upgrade value *only* applies to the change of Active Term to Start Term. If you set the End Term Equals field to Start Term, this will only apply for rows where Active Term does not equal either 0000 or 9999. The reason for this is that an End Term value of 9999 is the functional equivalent of an End Term set to Blank, while an End Term value of 0000 means that the Service Indicator is never in effect (End Term is infinitely in the past), which is the equivalent of a Start Term of 9999. If the End Term Equals field is set to Start Term for a Service Indicator, any assigned Service Indicator with an Active Term of 0000 or 9999 will end up with an assignment that is never in effect (that is, the Start Term and End Term will be equal to 0000, or the Start Term and End Term will be equal to 9999).

---

- Mapping codes

After analyzing your existing Service Indicator definitions and assignments, and using your knowledge of your institutional usage of Service Indicators, you will need to either map each existing code directly to a corresponding code in PeopleSoft 9 or split the code, in order to preserve the intention of assigned Service Indicators.

A direct map means exactly that; the row will be upgraded to PeopleSoft 9 with the same structure and impact associations as in your current release. You will still need to make decisions about End Term values and mapping of 9999 values.

Split mapping allows you to create new codes on the fly and reorganize the associated impacts so that term validity periods and 9999 values will retain their previous intended effect. Split mapping is only required for Service Indicator codes with mixed usage impacts.

For example, Service Indicator Code C01 has 3 impacts: LF001 (date-based), CENR (term- and date-based) and FAD (term-based). Because a direct map would mean that one or more of these impacts would lose its original assigned intention, it will be necessary to split this code and regroup the impacts in such a way that existing Service Indicator assignments are not functionally modified.

You could choose to split this Service Indicator Code into two codes: C01, with impacts LF001 and CENR, End Term Equals set to Blank, and the “Convert 9999 to” field set to 9999; and C02, with impact FAD, End Term Equals set to Start Term, and the “Convert 9999 to” field set to 0000. This would ensure that the impact CENR would still be interpreted correctly by Enrollment processing (an Active Term equivalent to first term would be converted to a Start Term with no defined End Term, and any value in the “Convert 9999 to” field means the impact is never in effect), and the impact FAD would still be interpreted correctly by Financial Aid processing (an Active Term equivalent to the *only* valid term would be converted to a Start Term with an equivalent End Term, and any value in the “Convert 9999 to” field means the impact is always in effect).



When these new codes are created, existing assigned Service Indicators will be upgraded accordingly, as will Service Indicator audit data. There is no requirement that previously existing codes be maintained. So, in the above example, you could split C01 into two completely new codes, C11 and C12.

---

**Note.** You should use caution when defining codes on the fly for splitting, so that you do not accidentally use the same code twice as a new code for two different previously existing codes, as this could cause significant confusion about the meaning and effect of a Service Indicator assignment. Also note that other than impact associations, all other values for a Service Indicator code definition (such as Positive/Negative, Reasons, and Comments) will be copied to the new code created in a split.

---

---

## Task Q-4: Understanding Service Indicator Audits

Because existing Service Indicator codes may be split into new codes and corresponding assignments, it is important that the Service Indicator audit record accurately reflects the new state of the Service Indicator assignments. Therefore, during the upgrade process, rows that are modified or created during the upgrade of the person and organization Service Indicator assignment tables will have a corresponding adjustment made in the Service Indicator audit table.

If a new row is added because of a split mapping, the following rules will be used:

- The Service Indicator date and time stamp (SRVC\_IND\_DTTM) will equal the value of the original row.
- The audit date and time stamp (AUDIT\_DTTM) will be set to the date and time that the upgrade was run.
- The OprID (AUDIT\_OPRID) will equal *UPG* to indicate that the row was added during the upgrade process.
- The audit action (AUDIT\_ACTN) will equal *Add*.
- Data for the original rows will remain in place.

Because audit data for Service Indicator assignments to external organizations was not kept in previous releases, the audit table will be populated with new rows for corresponding data in the Service Indicator Organization table (SCC\_SI\_ORG), using the first four rules listed above.

---

## Task Q-5: Understanding Service Indicator Display Security

Service Indicator Display is a new layer of security that allows for the role-based control of the display of Service Indicators on administrative pages. In order to maintain existing functionality, the upgrade process will not impose any Service Indicator display restrictions; that is, all users with access to administrative Service Indicator pages will have at least View access to all Service Indicator codes. If you want to impose restrictions on the display of Service Indicators to administrative users, you should review your institutional requirements and add Service Indicator Display security, as appropriate.



## APPENDIX R

# Upgrading the Content Provider Registry

This appendix discusses:

- Understanding the Content Provider Registry Upgrade
- Copying Your Enterprise Portal Database
- Upgrading Enterprise Portal PeopleTools
- Updating Registry Permission Lists
- Creating the Portal Project
- Comparing the Portal Project
- Reviewing the Portal Project
- Copying the Portal Project
- Copying Portal Project to Production
- Deleting Obsolete Folders
- Updating Registry Folder Permissions

---

## Understanding the Content Provider Registry Upgrade

You should perform this task if you use the Enterprise Portal 8.4 or higher running on PeopleTools 8.4x with the full navigation load access method. This means you do not use a single link to access your content provider database, but instead load some or all of the portal registry structures from the content provider database into your Enterprise Portal database. PeopleSoft refers to its application databases that contain the transaction content as Content Provider databases. Your Copy of Production database is your Content Provider database for this task.

When you upgrade a content provider database, the registry structures are updated, removed and added. These changes need to be copied to the Enterprise Portal database. This task will update the portal registry structures in your Enterprise Portal database to match what is in the Content Provider database. This is accomplished by the following:

- Upgrade the PeopleTools on a copy of the Enterprise Portal database.  
This allows a project compare to run between the Enterprise Portal and the Content Provider database.
- Create a portal project in the Enterprise Portal database containing all of the existing Content Provider registry structures.  
Copy the portal project (definition only) to the Content Provider database.

- Create a portal project in the Content Provider database containing all of the current Content Provider registry structures, then merge the project definition copied from the Enterprise Portal database into this project.

You will have a complete list of all registry structures for the Content Provider, including what is current and what should be deleted.

- Compare the complete list of registry structures in the Content Provider database to what exists in the Enterprise Portal, using project compare.

This marks the missing registry structures as delete and the updated or added registry structures as copy in the portal project definition.

- Copy the portal project from the Content Provider database to the Enterprise Portal database.

This deletes, updates, and adds registry structures to the Enterprise Portal database which syncs it up with what is current in the Content Provider database.

If you use Enterprise Portal 8 SP2, PeopleSoft recommends that you upgrade your Enterprise Portal to the latest available release.

If you do upgrade your Enterprise Portal database, you must be on PeopleTools 8.46 or higher.

---

**Note.** If you use Enterprise Portal 8.4 you *do not* need to upgrade to Enterprise Portal 8.8. You can still upgrade to PeopleTools 8.46.

---

See Enterprise Portal 8.1x – Managing Information Architecture on PeopleSoft Customer Connection (Support, Documentation, Documentation Updates, Enterprise, Portal Products, Enterprise Portal) for additional information on this topic.

In this appendix, you load your new Portal Registry definitions from your Copy of Production database to a copy of your Enterprise Portal database.

---

**Note.** You must complete the tasks in the appendix for each of your separately installed Enterprise Portal databases that correspond to one of the four Portal Registry definitions: EMPLOYEE, CUSTOMER, SUPPLIER, and PARTNER. If your installed Enterprise Portal uses all the registries, then complete this task for each of the portal registries using the same copy of the single Enterprise Portal database.

---

In the first task of this appendix, you create a copy of your Enterprise Portal database. You use this copy for all subsequent steps for the initial and test Move to Production upgrade passes. For the final Move to Production, do not make a copy. Instead perform the steps on the production Enterprise Portal database.

This document uses the term “target Enterprise Portal database” to refer to the Enterprise Portal database used in the upgrade steps. Use the table below to determine the correct copy of your Enterprise Portal database for each pass:

Upgrade Pass	Target Enterprise Portal Database
Initial pass	Copy of the Enterprise Portal database
Test Move to Production	Copy of the Enterprise Portal database
Final Move to Production	Enterprise Portal production database

---

## Task R-1: Copying Your Enterprise Portal Database

You initially upgrade the Content Provider registry on a copy of your Enterprise Portal database, then test the results of the upgrade. During your test Move to Production, you perform this task against another Copy of the Enterprise Portal.

Create a copy of your current Enterprise Portal production database now. Use this database as your target Enterprise Portal database.

---

**Note.** During your final Move to Production, you copy the registry definitions directly to your Enterprise Portal production database. Therefore, you do not need to execute this step during your final Move to Production.

---

---

## Task R-2: Upgrading Enterprise Portal PeopleTools

During the initial upgrade pass, your Enterprise Portal database must run on the same PeopleTools release level as your Copy of Production database so that you can do the compare step. Because you do not need to run the compare step during your Move to Production passes, you can skip this task during Move to Production passes.

If the release level of PeopleTools on your target Enterprise Portal database is not the same as your Copy of Production database release level, upgrade your PeopleTools now.

See PeopleSoft Customer Connection (Implement Optimize + Upgrade, Upgrade Guide, Upgrade Documentation and Software, Upgrade Documentation and Scripts, Release, Enterprise, PeopleTools).

---

## Task R-3: Updating Registry Permission Lists

This section discusses:

- Understanding Registry Permission Lists Updates
- Updating Portal Registry
- Deleting the Database Cache

### Understanding Registry Permission Lists Updates

This task applies only to the initial upgrade pass.

Earlier in this upgrade you copied portal registry data from the Demo database to your Copy of Production database. You must update this registry data to include your permission list changes. After updating the portal registry permission lists, delete the database cache.

This process takes between a few minutes and a few hours, depending on the volume of the portal data.

---

**Note.** The user ID that invokes this process must have the security role Portal Administrator, or the process may terminate with an abend.

---

---

**Note.** You must have a process scheduler started for your Copy of Production database.

---

## Task R-3-1: Updating Portal Registry

Follow the steps below to update your portal registry permission lists.

To update the portal registry permission lists:

1. On your Copy of Production database, select PeopleTools, Portal, Portal Security Sync.
2. Select the Add a New Value tab.
3. Add a run control as follows:
  - a. Enter a value for the run control ID. The run control ID is *SECURITY\_SYNC\_XXXX*, where *XXXX* represents the portal registry name (EMPLOYEE, CUSTOMER, SUPPLIER, or PARTNER).
  - b. Click Add.
4. Enter a value for the portal name.

This value must match the portal registry name you used to replace the *XXXX* in the run control ID.
5. Click Save.
6. Click Run.
7. Set up the process scheduler information and click OK.
8. Click the Process Monitor link to view the progress of the process.

## Task R-3-2: Deleting the Database Cache

Follow the steps below to delete the database cache.

To delete the database cache:

1. Delete the Copy of Production database application server cache.
2. Stop and restart the Copy of Production database web server service.

---

## Task R-4: Creating the Portal Project

This section discusses:

- Understanding Portal Project Creation
- Creating the Target Enterprise Portal Project
- Cleaning the Target Enterprise Portal Project
- Deleting the Target Enterprise Portal Database Cache
- Copying the Target Enterprise Portal Project Definition
- Creating the Copy of Production Portal Project
- Cleaning the Copy of Production Portal Project
- Deleting the Copy of Production Database Cache

## Understanding Portal Project Creation

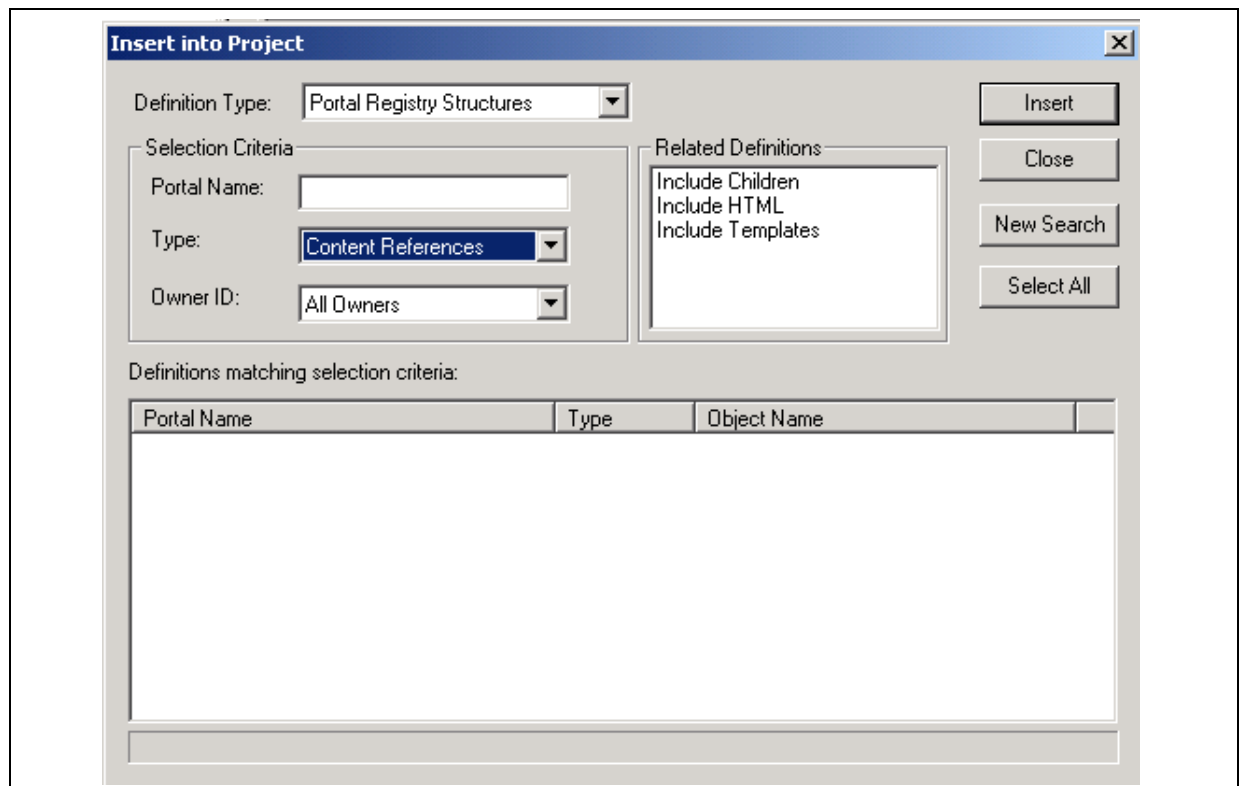
This task applies only to the initial upgrade pass. In this task, you create and modify a project on your target Enterprise Portal database. Then you copy the project definition to the Copy of Production database, where you further modify the project.

### Task R-4-1: Creating the Target Enterprise Portal Project

Follow the steps below to create the target Enterprise Portal project.

To create the target Enterprise Portal project:

1. Launch Application Designer and sign on to your target Enterprise Portal database.
2. Select Insert, Definitions into Project...
3. Select the following values:
  - a. In the Definition Type field, select *Portal Registry Structures*.
  - b. Leave the Portal Name field blank.
  - c. In the Owner ID field, select *All Owners*.
  - d. Do not select any values in the Related Definitions field.



Insert into Project dialog box

4. Click Insert.
5. Click Select All, and then click Insert again.
6. Click Close.
7. From Application Designer, select File, Save Project As....

8. Enter the project name *PORTAL\_PA84X\_REGISTRY*.
9. Close Application Designer.

## Task R-4-2: Cleaning the Target Enterprise Portal Project

In this step, you clean the target Enterprise Portal Project so that it contains only the existing Content Provider registry structure content references.

To clean the target Enterprise Portal project:

1. In your Enterprise Portal database, select PeopleTools, Portal, Portal Utilities, Clean Portal Project.

---

**Warning!** Do not follow the instructions on the Clean Portal Project page. Instead, follow the instructions below.

---

2. Add the run control ID *CLEAN\_PORTAL\_XXXXXXXX* where *XXXXXXXX* represents the portal definition name: *EMPLOYEE*, *CUSTOMER*, *SUPPLIER* or *PARTNER* for example.
3. In the Project Name field, enter the project name *PORTAL\_PA84X\_REGISTRY*.
4. Enter a value in the Portal Name field; *EMPLOYEE* for example.
5. Enter a value in the Content Provider Name field; *CRM* for example.

---

**Note.** Before running the Clean Portal Project you must enter the node URI text for the message node that you selected.

---

6. Select *Full Navigation*.
7. Click Save.
8. Click Run.
9. Set up the Process Scheduler information and click OK.
10. Select the Process Monitor link to view the progress of the process.

## Task R-4-3: Deleting the Target Enterprise Portal Database Cache

In this step, you delete the target Enterprise Portal database cache.

To delete the target Enterprise Portal database cache:

1. On your target Enterprise Portal database, launch Configuration Manager.
2. On the Startup tab, click Purge Cache Directories.
3. Select the target Enterprise Portal database name.
4. Click Delete.
5. Click OK.
6. Click Close.
7. Click OK to close Configuration Manager.



## Task R-4-4: Copying the Target Enterprise Portal Project Definition

In this step, you copy the target Enterprise Portal project definition to your Copy of Production database.

To copy the target Enterprise Portal project definition:

1. Using PeopleSoft Data Mover, sign on to your target Enterprise Portal database.
2. Run the following Data Mover script:

```
PS_HOME\SCRIPTS\UVUPX10E.dms
```

3. Close Data Mover.
4. Using PeopleSoft Data Mover, sign on to the Copy of Production database.
5. Run the following Data Mover script:

```
PS_HOME\SCRIPTS\UVUPX10I.dms
```

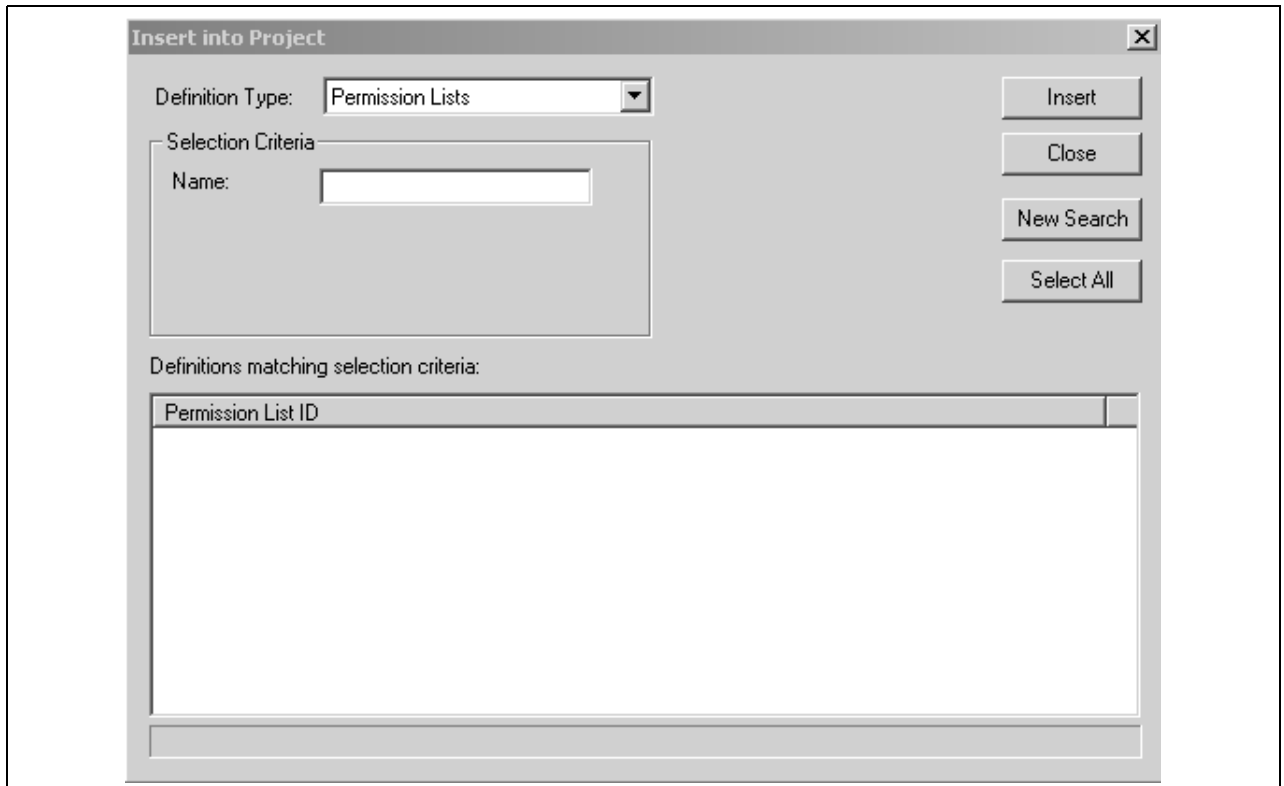
6. Close Data Mover.

## Task R-4-5: Creating the Copy of Production Portal Project

Create a project containing all Portal Registry data on your Copy of Production database.

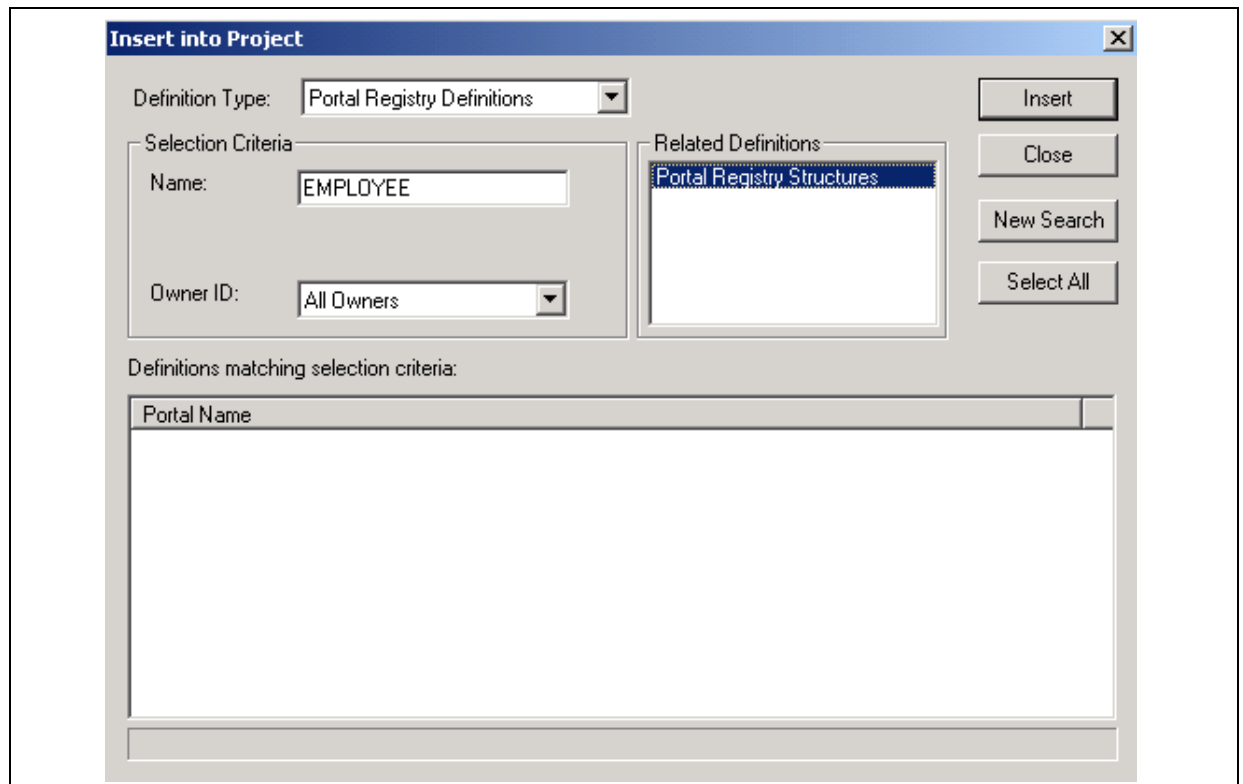
To create the Copy of Production Portal project:

1. Launch Application Designer and sign on to your Copy of Production database.
2. Select Insert, Definitions into Project....
3. In the Definition Type field, select *Permission Lists*.



Insert into Project dialog box: Definition Type Permission Lists

4. Click Insert.
5. Click Select All, and then click Insert again.
6. Select the following values:
  - a. In the Definition Type field, select *Portal Registry Definitions*.
  - b. In the Name field, enter the Enterprise Portal database's default portal name (EMPLOYEE, CUSTOMER, SUPPLIER or PARTNER).
  - c. In the Owner ID field, select *All Owners*.
  - d. In the Related Definitions field, select *Portal Registry Structures*.



Insert into Project dialog box with Portal Registry Structures selected

7. Click Insert.
8. Click Select All, then click Insert again.
9. Click Close.
10. From Application Designer, select File, Save Project As....
11. Enter the appropriate new project name from the table below. This project is referred to as the Portal Project:

Portal Name	Project Name
EMPLOYEE	PORTAL_APP84X_EMPLOYEE
CUSTOMER	PORTAL_APP84X_CUSTOMER
PARTNER	PORTAL_APP84X_PARTNER
SUPPLIER	PORTAL_APP84X_SUPPLIER

12. Click OK.
13. From Application Designer, select File, Merge Projects...
14. Enter the project name *PORTAL\_PA84X\_REGISTRY*.  
This merges the objects from the PORTAL\_PA84XREGISTRY project into your newly created Portal Project.
15. Select File, Save Project to save the updated Portal Project.
16. Close Application Designer.

## Task R-4-6: Cleaning the Copy of Production Portal Project

In this step, you clean the Copy of Production Portal project so that it contains only the Content Provider registry data.

---

**Important!** Before using the Copy of Production Portal project, you must run the Clean Portal Project on the Copy of Production database. Follow the directions on the Clean Portal Project Page.

---

To clean the Copy of Production Portal project:

1. In your Copy of Production database, select PeopleTools, Portal, Portal Utilities, Clean Portal Project.
2. Add the run control ID, *CLEAN\_PORTAL\_XXXXXXX*, where *XXXXXXX* represents the portal definition name; *EMPLOYEE*, *CUSTOMER*, *SUPPLIER*, or *PARTNER*, for example.
3. In the Project Name field, enter the Portal Project name that you created in the Creating the Copy of Production Portal Project step (*PORTAL\_APP84X\_[your portal name here]*).
4. Enter a value in the Portal Name field; *EMPLOYEE*, for example.
5. Enter a value in the Content Provider Name field; *CRM*, for example.

---

**Important!** Before running the Clean Portal Project, you must enter the Node URI text for the Message Node you selected.

---

6. Select *Full Navigation*.
7. Click Save.
8. Click Run.
9. Set up the Process Scheduler information and click OK.
10. Select the Process Monitor link to view the progress of the process.

## Task R-4-7: Deleting the Copy of Production Database Cache

In this step, you delete the Copy of Production database cache.

To delete the Copy of Production database cache:

1. On your Copy of Production database, start Configuration Manager.
2. On the Startup tab, click Purge Cache Directories.
3. Select the Copy of Production database name.
4. Click Delete.
5. Click OK.
6. Click Close.
7. Click OK to close Configuration Manager.

---

## Task R-5: Comparing the Portal Project

This task applies only to the initial upgrade pass.

In this step, you compare the Portal project that you created in the previous step and then review the compare results. This will enable you to adjust the portal project as necessary before copying it into the Enterprise Portal database.

To compare the Portal Project:

1. Launch Application Designer and sign on to your Copy of Production database.
2. Select Tools, Compare and Report....
3. Enter the Portal Project name that you specified in the Creating the Copy of Production Portal Project step (PORTAL\_APP84X\_[your portal name here]).
4. Enter the database name of your target Enterprise Portal database, and the user ID and password.
5. Click the Options button.
6. In the Compare Type field, select *Project*, and click OK.
7. Select all object types and click OK.
8. Close Application Designer.

---

## Task R-6: Reviewing the Portal Project

This task applies only to the initial upgrade pass.

Review the Portal project (PORTAL\_APP84X\_[your portal name here]) on the Copy of Production database, looking for customizations that you have applied to your database. Object definitions that you changed have *\*Changed* or *\*Unchanged* in the Target column of the compare report. The \* indicates that the change was not made by PeopleSoft. Review each of these objects carefully. If PeopleSoft delivered the object, the Source column of the report will read *Changed*. Note the changes that you made to the object. After you complete the upgrade, when you test the system, you can decide whether you still need the customization. You can reapply the customization at that time.

See Appendix: “Using the Comparison Process.”

---

## Task R-7: Copying the Portal Project

This section discusses:

- Understanding Portal Project Copying
- Copying the Portal Project
- Deleting the Enterprise Portal Database Cache

### Understanding Portal Project Copying

This task applies only to the initial upgrade pass.

In this step, you copy the project from your Copy of Production database to your target Enterprise Portal database.

## Task R-7-1: Copying the Portal Project

Follow the steps below to copy the Portal Project to the Enterprise Portal database.

---

**Important!** Before exporting the Portal Project from the Content Provider database, you must successfully clean the Copy of Production Portal Project. If you proceed with this step without cleaning the project, you will overwrite critical Enterprise Portal data.

---

See Creating the Portal Project, Cleaning the Copy of Production Portal Project.

To copy the Portal Project:

1. Launch Application Designer and sign on to your Copy of Production database.
2. Select File, Open...
3. In the Definition field, select *Project* and click Open.
4. Highlight the newly created Portal Project name (PORTAL\_APP84X\_[your portal name]) and click Open again.
5. Select Tools, Copy Project, To Database...
6. Enter the name of your target Enterprise Portal database, and the user ID and password.
7. Click Select All.
8. Click Copy.

This may take a few minutes.

9. Close Application Designer.

---

**Note.** You do not need to create or alter any records or views.

---

## Task R-7-2: Deleting the Enterprise Portal Database Cache

In this step, you delete the Enterprise Portal database cache.

To delete the Enterprise Portal database cache:

1. Delete the target Enterprise Portal database application server cache.
2. Stop and restart the target Enterprise Portal database web server service.

---

## Task R-8: Copying Portal Project to Production

This section discusses:

- Understanding Portal Project to Production Copying
- Copying the Portal Project to File
- Copying the Portal Project from File
- Deleting the Enterprise Portal Database Cache Again

## Understanding Portal Project to Production Copying

You must perform this step during both your test and final Move to Production upgrade passes.

### Task R-8-1: Copying the Portal Project to File

Follow the steps below to copy the Portal Project to file.

---

**Note.** If your Copy of Production and target Enterprise Portal databases run on the same PeopleTools release and database platform, you can copy the project directly to the target Enterprise Portal database from within the Copy of Production Application Designer and skip the rest of this step.

---

To copy the Portal Project to file:

1. Launch Application Designer and sign on to your Copy Production database.
2. Select File, Open....
3. In the Definition field, select *Project* and then click Open.
4. Highlight the newly created Portal Project name (PORTAL\_APP84X\_[your portal name]) and click Open again.
5. Select Tools, Copy Project, To File....
6. Click the Browse button for the Export Directory.
7. Select a temporary directory and then click OK.
8. Click Select All.
9. Click Copy.  
This may take a few minutes.
10. Close Application Designer.

### Task R-8-2: Copying the Portal Project from File

In this step, you copy the Portal Project from file.

To copy the Portal Project from file:

1. Launch Application Designer and sign on to your target Enterprise Portal database.
2. Select Tools, Copy Project, From File....
3. Browse to the Copy of Production database server's temporary directory.  
If you cannot access the Copy of Production database server's temporary directory, then copy the Portal Project folder and files from the temporary directory to the target Enterprise Portal database server's *PS\_HOME\PROJECTS* directory, and browse to that directory.
4. Select the Portal Project name that you just copied to file in the previous step.
5. Click Open.
6. Click Select All.
7. Set the project language options as follows:
  - a. Click Options.
  - b. In the Copy Options tab, select *English*, and *COMMON*.

- c. If your Enterprise Portal database is a multi-language database, then also select the languages that you have installed on your Enterprise Portal database.
  - d. Click OK.
8. Click Copy.
  9. Select the Upgrade tab and view the Output window.  
All objects should have copied successfully.
  10. Close Application Designer.

---

**Note.** After the copy, you do not need to create or alter any records or views on the target Enterprise Portal database.

---

## Task R-8-3: Deleting the Enterprise Portal Database Cache Again

In this step, you delete the Enterprise Portal database cache.

To delete the Enterprise Portal database cache:

1. Delete the target Enterprise Portal database's application server cache.
2. Stop and restart the target Enterprise Portal database web server service.

---

## Task R-9: Deleting Obsolete Folders

This section discusses:

- Understanding Obsolete Folder Deletion
- Deleting Obsolete Folders on Enterprise Portal 8.4
- Deleting Obsolete Folders on Enterprise Portal 8.8

### Understanding Obsolete Folder Deletion

This task applies to all upgrade passes: Initial, Test Move to Production, and Final Move to Production.

In this step, you delete folders on your target Enterprise Portal database that the Portal Registry Structures no longer reference. The process that you run depends on your version of Enterprise Portal.

### Task R-9-1: Deleting Obsolete Folders on Enterprise Portal 8.4

Follow this procedure to delete obsolete folders on Enterprise Portal 8.4.

To delete obsolete folders on Enterprise Portal 8.4:

1. Using PeopleSoft Data Mover, sign on to your target Enterprise Portal database.
2. Run the following Data Mover script, located in the Enterprise Portal PS\_HOME\SCRIPTS directory:

```
PORTAL_REG_FOLDER_DEL.DMS
```

3. Close Data Mover.



## Task R-9-2: Deleting Obsolete Folders on Enterprise Portal 8.8

Follow this procedure to delete obsolete folders on Enterprise Portal 8.8 or higher.

To delete obsolete folders on Enterprise Portal 8.8 or higher:

1. On your target Enterprise Portal database, navigate accordingly:
  - a. For Enterprise Portal 8.8: Portal Administration, Navigation, Run Folder Cleanup.
  - b. For Enterprise Portal 8.9 or higher: Portal Administration, Navigation, Delete Empty Folders.
2. Add a run control as follows:
  - a. Enter a value for the run control ID. The run control ID is *FOLDER\_CLEAN\_XXXX*, where *XXXX* represents the portal registry name (EMPLOYEE, CUSTOMER, PARTNER, or SUPPLIER).
  - b. Click Add.
3. Enter a value in the Portal Name field.

This value must match the portal registry name that you used to replace *XXXX* in the run control ID (EMPLOYEE, CUSTOMER, PARTNER, or SUPPLIER).
4. Click Save.
5. Click Run.
6. Set up the process scheduler information and click OK.
7. Click the Process Monitor link to view the progress of the process.

---

## Task R-10: Updating Registry Folder Permissions

This section discusses:

- Understanding Registry Folder Permissions Updates
- Updating the Enterprise Portal Registry Folder Permissions
- Deleting the Enterprise Portal Cache

### Understanding Registry Folder Permissions Updates

This task applies to all upgrade passes: Initial, Test Move to Production, and Final Move to Production.

Portal data from different Content Provider databases may share a common portal folder. After copying the registry projects, you must update the folder permissions to reflect the changes. After you update the folder permissions, you must delete the target Enterprise Portal database cache files to propagate the changes.

### Task R-10-1: Updating the Enterprise Portal Registry Folder Permissions

Follow this procedure to update your Enterprise Portal registry folder permissions.

---

**Note.** This process will take between a few minutes to a few hours, depending on the volume of portal data. The user ID that invokes this process must have the security role Portal Administrator, or the process may terminate with an abend.

---

To update the Enterprise Portal folder permissions:

1. On your target Enterprise Portal database, select PeopleTools, Portal, Portal Security Sync.
2. Add a run control as follows:
  - a. Enter a value for the run control ID.  
The run control ID is *SECURITY\_SYNC\_XXXX*, where *XXXX* represents the portal registry name (EMPLOYEE, CUSTOMER, PARTNER, or SUPPLIER).
  - b. Click Add.
3. Enter a value in the Portal Name field.  
This value must match the portal registry name that you used to replace *XXXX* in the run control ID (EMPLOYEE, CUSTOMER, PARTNER, or SUPPLIER).
4. Click Save.
5. Click Run.
6. Set up the process scheduler information and click OK.
7. Click the Process Monitor link to view the progress of the process.

## Task R-10-2: Deleting the Enterprise Portal Cache

In this step delete the Enterprise Portal cache.

To delete the Enterprise Portal cache:

1. Delete the target Enterprise Portal database application server cache.
2. Stop and restart the target Enterprise Portal database web server service.

## APPENDIX S

# Upgrading with Tax Updates

---

## Upgrading with Tax Updates

Your HRMS Demo database was delivered with a predetermined level of Tax Updates applied. This Tax Update level is listed in the task, “Understanding Your Upgrade.” Oracle recommends you run your upgrade using your Copy of Production at the same Tax Update level as the HRMS Demo database. Your Copy of Production database may be in one of three states relative to the Demo database.

- You will need to update the delivered HRMS Demo database with all tax updates through the latest tax update on your Copy of Production before you start your initial upgrade pass. This is required so you will not lose any of the more current Tax Updates. Do not apply any more updates to your production system until your upgrade is complete and you go live on the new PeopleSoft release. If you apply additional updates to your production system while running the upgrade passes, you will lose those updates during your Move to Production pass. You will then need to reapply the tax updates that were applied to your production since your Initial upgrade pass.
- The Copy of Production database may be current with the Tax Update from the Demo database applied (but not beyond). You do not need to apply any tax updates.
- The Copy of Production database may not be current with the Tax Update from the Demo database. Oracle strongly recommends that you apply all of the Tax Updates to bring your Copy of Production database to the same Tax Update level as your HRMS Demo database. This will assure that your tax information will be correct when you have completed your upgrade. If this is not possible, be aware of the following potential issues and act accordingly:

- To get your Tax Tables up to date, you will need to compare data in the Tax Tables from your Copy of Production to the HRMS Demo database. Any needed updates from the Demo Database should be applied to the Copy of Production database manually. The tables involved are as follows:

PS\_CAN\_TAX\_TBL  
 PS\_CAN\_TAX\_CITLUMP  
 PS\_CAN\_TAX\_QITLUMP  
 PS\_CAN\_TAX\_CITRATE  
 PS\_CAN\_TAX\_QITRATE  
 PS\_CAN\_TAX\_PROV  
 PS\_CAN\_TAX\_PROVNCT  
 PS\_CAN\_TAX\_PROVTHR

PS\_GARN\_RULE\_TBL  
 PS\_GARN\_EXEMPT\_TBL  
 PS\_GARN\_OPERANDS  
 PS\_GARN\_DE\_DEFN  
 PS\_GARN\_DE\_DED  
 PS\_GARN\_PRORATE\_RL

PS\_STATE\_TAX\_TBL  
 PS\_STATE\_TAXRT\_TBL  
 PS\_ST\_OTH\_TAX\_TBL  
 PS\_STTAX\_RECIP\_TBL  
 PS\_LOCTX\_RECIP\_TBL  
 PS\_TAXGR\_DEFN\_TBL  
 PS\_TAXGR\_BASE\_TBL  
 PS\_TAXGR\_CMPNT\_TBL  
 PS\_LOCAL\_TAX\_TBL  
 PS\_LOCAL\_TAX\_TBL2  
 PS\_SWT\_MARSTAT\_TBL  
 PS\_LCLWK\_TXRCP\_TBL

---

**Note.** Garnishments Rules tables GARN\_RULE\_TBL, GARN\_EXEMPT\_TBL, and GARN\_OPERANDS have a different structure in this release and will be part of data conversion. We will deliver all garnishments rules maintained by us during the upgrade, so there is no need to manually update these tables to bring them to the Tax Update level of the Demo database.

---

See Appendix: “Understanding Garnishments Changes.”

- You should also review the Tax Update documentation to guide you on specific Tax Updates. This documentation is available on PeopleSoft Customer Connection.

# APPENDIX T

## Using Data Conversion Utilities

This appendix discusses:

- Understanding Data Conversion Utilities
- Using the Upgrade Driver Program
- Using the Upgrade Drivers Page
- Reviewing the Data Conversion Report

---

### Understanding Data Conversion Utilities

This appendix contains information regarding the Application Engine program UPG\_DATACONV and the PS\_UPG\_DATACONV table.

---

### Task T-1: Using the Upgrade Driver Program

In order to run all data conversions in the correct sequence, Oracle has provided the Application Engine program UPG\_DATACONV and the PS\_UPG\_DATACONV table. This program runs the Application Engine sections defined in the table PS\_UPG\_DATACONV. The PS\_UPG\_DATACONV table contains a list of all of the Application Engine sections that you need to run and in what sequence they should be run.

---

### Task T-2: Using the Upgrade Drivers Page

This section discusses:

- Understanding Upgrade Drivers Page
- Accessing the Upgrade Drivers Page
- Adding New Upgrade Drivers Section Page
- Inactivating Upgrade Drivers Section

### Understanding Upgrade Drivers Page

Before you run data conversion, you may need to change what the Upgrade Driver program runs. You can add, remove, or deactivate Application Engine sections through the Upgrade Drivers page.

You do not have an active portal on your Copy of Production during data conversion, so you need to view and update the Data Conversion Definitions on your Demo database and then copy the updated data to your Copy of Production database.

## Task T-2-1: Accessing the Upgrade Drivers Page

To access the Upgrade Drivers page:

1. From your browser, sign on to the Demo database.
2. Select Set Up HRMS, Upgrade, Data Conversion.
3. Enter your upgrade path:

HC80 for groups 1 to 3 and 6 to 8

CS80 for groups 10 and 11

4. Click Search.

You are now on the Upgrade Drivers page. The following are descriptions for each section of the Upgrade Drivers page.

Upgrade Drivers									
Customize   Find   View All   First 1-25 of 86 Last									
Upgrade Path	Program Name	Group #	Section	Sequence	Active Flag	Description	Comments		
CR80	UPG_CDM	1	CDMA010	10	Active	General Preparation	Comments	+	-
CR80	UPG_CDM	1	CDMX140	20	Active	Upgrade Basic Data Tables	Comments	+	-
CR80	UPG_CP	2	CPA00	100	Active	Upgrade Constraint	Comments	+	-
CR80	UPG_CP	2	CPA01	105	Active	Upgrade User Cd Detl	Comments	+	-

Upgrade Drivers page

- Upgrade Path. This field contains the upgrade path on which the section will be run.
- Program Name. This is the Application Engine program that contains the section.
- Group #. This is the group number. All sections with the same group number will be run during the same run of the UPG\_DATACONV Application Engine program.
- Section. This is the section that will be called from the UPG\_DATACONV Application Engine program.
- Sequence. This is the order in which the sections will be called during the run of UPG\_DATACONV for the group number.
- Active Flag. This field determines whether the section will be run. If the value of this field is *Active*, the section will be run. If the value is *Inactive*, it will not be run. If you need to remove a section, change the value in this field to *Inactive*.
- Description.
- Comments.

## Task T-2-2: Adding New Upgrade Drivers Section Page

Follow the instructions below to add a new section to the Upgrade Drivers page.

**Note.** To add a new section, the Application Engine program and section must exist on the Demo database.

To add a new section to the Upgrade Drivers page:

1. From your browser, sign on to the Demo database.
2. Select Set Up HRMS, Upgrade, Data Conversion.
3. Select Add a New Value.
4. Click Add.
5. Enter values for Upgrade Path and Program Name.
6. Enter a value for Group #.

---

**Note.** Each group number corresponds to a data conversion step in the Change Assistant template. If you select a group number that already exists in the PS\_UPG\_DATACONV table, your section will be executed when Change Assistant runs the data conversion step that corresponds to the group number you selected. Alternatively, if you assign a group number to your new section that does not already exist in PS\_UPG\_DATACONV, you must add a new step to your Change Assistant template. The new template step will have the same properties as the other data conversion steps, except for the group number specified in the step properties Parameters box.

---

7. Enter values for Section and Sequence.  
The Description and Comments fields are optional.
8. Click Save.
9. When you have completed all changes, sign on to your Demo database using Data Mover and run the following script to export the updated data conversion data:

```
DLUPX03E.DMS
```

10. Sign on to your Copy of Production database using Data Mover and run the following script to load the updated data conversion data:

```
DLUPX03I.DMS
```

See the Enterprise PeopleTools PeopleBook: PeopleSoft Software Updates for your new release, Appendix: "Using a Change Assistant Template."

## Task T-2-3: Inactivating Upgrade Drivers Section

Follow the instructions below to deactivate a section on the Upgrade Drivers page. Once deactivated, the section will not run as part of data conversion.

To inactivate a section on the Upgrade Drivers page:

1. From your browser, sign on to the Demo database.
2. Select Set Up HRMS, Upgrade, Data Conversion.
3. Enter your upgrade path:  
  
HC80 for groups 1 to 3 and 6 to 8  
CS80 for groups 10 and 11
4. Click Search.
5. Find the row with the Program Name and Section you want to remove and change the value of the Active Flag field to *Inactive*.
6. Click Save.

7. When you have completed all changes, sign on to your Demo database using Data Mover and run the following script to export the updated data conversion data:

```
DLUPX03E.DMS
```

8. Sign on to your Copy of Production database using Data Mover and run the following script to load the updated data conversion data:

```
DLUPX03I.DMS
```

---

## Task T-3: Reviewing the Data Conversion Report

Each of the upgrade data conversion sections contains comments that describe the processing performed by the section. PeopleSoft has delivered an SQR to list all of these comments by the group and sequence numbers that determine how they run. The name of this report is UDATAACNV.

---

**Note.** During the Campus Solutions upgrade you run two reports to list all data conversion sections. You run data conversion for the HR core as well as data conversion sections specific to Campus Solutions.

---

To run UDATAACNV:

1. Using SQRW, run SQR UDATAACNV on your Copy of Production database.
2. When prompted for Upgrade Path, enter:

```
HC80 for a report on groups 1 to 8  
CS80 for a report on groups 10 and 11
```

3. When prompted for Group Number, enter the two-digit group number to report on, or enter 0 to see the comments for all groups.



## APPENDIX U

# Using the Comparison Process

This appendix discusses:

- Understanding the Comparison Process
- Understanding Upgrade Compare Reports

---

## Task U-1: Understanding the Comparison Process

This section discusses:

- Reviewing Source and Target Columns
- Reviewing the Action Column
- Reviewing the Upgrade Column
- Putting it All Together

During the upgrade you run a compare process and then review the resulting reports. The compare process first compares every property of an object definition on the source database to the properties of object definitions on the target database. The PeopleSoft system tracks object changes using the contents of the PSRELEASE table, and the value of two fields, LASTUPDDTTM, and LASTUPDOPRID, used in the PeopleTools tables.

- The PSRELEASE table maintains the Comparison Release Level. This table contains rows of data for every release level at which the database has ever existed. The first column in this table, RELEASEDTTM, contains a date/time stamp identifying when each release level was “stamped.” The second column, RELEASELABEL, identifies the release level. The format of a release label is *M XX.XX.XX.YYY*, where *M* is the market code, *XX* is an integer from 0 to 99, and *YYY* is an integer from 0 to 999. A release label has two parts: the PeopleSoft release number (*M XX.XX.XX*) and the customer release number (*YYY*). Each time you customize your production database, you can stamp it with a new customer release level to help you track your changes over time. You should not change any portion of the PeopleSoft release number unless specifically instructed to do so.
- The LASTUPDDTTM field in our *PSobjectDEFN* tables—such as PSRECDEFN, PSPNLDEFN, and so on—stores a date/time stamp of when each object was last modified.
- The LASTUPDOPRID field stores the operator ID of the user who made the modification. If PeopleSoft made the modification, the proprietary ID *PPLSOFT* is used.

---

**Note.** Maintain Security prevents you from creating an operator named PPLSOFT.

---

If an object definition is defined differently in the source database than in the target, the compare process will check to see whether either object definition has changed since the comparison release. If the object's LASTUPDDTTM value is greater than the RELEASEDTTM value for the comparison release level (stored in PSRELEASE), the object has changed. If the object's LASTUPDDTTM value is equal to or less

than RELEASEDTM, the object has not changed (since the comparison release). Whether the compared object has changed or not, if it has *ever* been changed prior to the comparison release by someone other than PeopleSoft (LASTUPDOPRID <> 'PPLSOFT'), the object is identified as a customization.

After you run a compare report, you see the following information when you open an object type in the upgrade project from the Upgrade Tab of Application Designer. This is called the Application Designer Upgrade Definition window.

## Task U-1-1: Reviewing Source and Target Columns

The status of each object is reported as it appears on the Source database and the Target database. The following table explains the various status types:

Status Type	Definition
Unknown	The object has not been compared. This is the default status for all objects inserted manually into a project and the permanent status of all non-comparison objects.
Absent	The object was found in the other database, but not in this one. When upgrading to a new PeopleSoft release, all of our new objects should have Absent status in the Target database and all of your new objects should have Absent status in the Source database.
Changed	The object has been compared, its LASTUPDOPRID value is <i>PPLSOFT</i> , and its LASTUPDTIME value is greater than the date/time stamp of the comparison release database. In other words, PeopleSoft modified the object since the comparison release.
Unchanged	The object has been compared, its LASTUPDOPRID value is <i>PPLSOFT</i> , and its LASTUPDTIME value is less than or equal to the date/time stamp of the comparison release database. In other words, PeopleSoft last modified the object prior to the comparison release.
*Changed	The object has been compared, its LASTUPDOPRID value is not <i>PPLSOFT</i> , and its LASTUPDTIME value is greater than the date/time stamp of the comparison release database. In this case, the customer has modified the object since the comparison release.

Status Type	Definition
*Unchanged	The object has been compared, its LASTUPDOPRID value is not <i>PPLSOFT</i> , and its LASTUPDTIME value is less than or equal to the date/time stamp of the comparison release database. In this case, the customer last modified the object prior to the comparison release.
Same	The object has been compared and is defined as the same in both databases. When an object in one database has this status, so will its counterpart in the other database. This status would never be seen when performing a database comparison because in that case, the project is only populated with objects defined differently. However, it can occur when performing a project comparison because in a project comparison, the project contents are static; the project is not repopulated based on the comparison results.

## Task U-1-2: Reviewing the Action Column

The default actions for each object that you compared are reported in the Action column. The compare sets the action column based on what you need to do to make the Target database consistent with the Source. You should not change these actions. You can decide whether or not to accept each action by setting the Upgrade value. Action types include:

Action Type	Definition
Copy	Object will be added to the Target database
Copy Prop (Records and Fields only)	Object will be added to the Target database
Delete	Object will be deleted from the Target database.
None	No action will be taken on this object.

The PeopleSoft system assigns one of these action types to every object in a comparison project and in the compare reports. However, these actions are not necessarily carried out during the copy process. The value of the Upgrade column for each object makes that determination.

## Task U-1-3: Reviewing the Upgrade Column

The Upgrade values for each object – YES or NO – determine whether the object action will be carried out during the copy process. The upgrade orientation you assign during the compare process determines these settings. You can orient the Upgrade to keep PeopleSoft changes or to retain your changes in the target database. Whichever orientation you choose, you will still have the option to set each Upgrade value individually before launching the copy process.

You may find that after the compare process, your project contains objects that show up as Unchanged on the Demo database and Changed on the Copy of Production and the Upgrade column is not checked. What this status combination means is that the PeopleSoft object on your Copy of Production was changed more recently than on the Demo database. In these instances, PeopleSoft recommends that you accept the Demo database version of the object.

## Task U-1-4: Putting it All Together

The following chart summarizes every possible Status, Action, and Upgrade value that could be set by the compare process to a single object:

Source Status	Target Status	Action	PeopleSoft Vanilla	Keep Customizations
(Any)	Absent	COPY	YES	YES
Absent	Changed or Unchanged	DELETE	YES	YES
	Changed* or Unchanged*	DELETE	NO	NO
Changed	Changed or Unchanged	COPY	YES	YES
	Changed* or Unchanged*	COPY	YES	NO
Unchanged	Changed	COPY	NO	NO
	Unchanged	COPY	YES	YES
	Changed* or Unchanged*	COPY	YES	NO
Changed*	Changed or Unchanged	COPY	NO	YES
	Changed* or Unchanged*	COPY	YES	YES
Unchanged*	Changed or Unchanged	COPY	NO	YES
	Changed*	COPY	NO	NO
	Unchanged*	COPY	YES	YES

---

## Task U-2: Understanding Upgrade Compare Reports

This section discusses:

- Reviewing Report Columns
- Using Reports

When you run the compare process, it creates reports to help you understand what objects differ between the Source and Target databases, and how they differ. If you have documentation of your database modifications, you should retrieve it before reviewing these reports. This will help you understand how the Target objects

have changed and enable you to better compare the Target version of the object with the Source version. If you are upgrading to a new PeopleSoft release, you should also review the Release Notes for your product. These notes will identify and explain object changes in the New Release Demo database.

Upgrade reports can be a little intimidating at first glance, until you understand what data you are looking for and how best to use it. This section includes information to help you use the reports.

## Task U-2-1: Reviewing Report Columns

For the most part, the columns in upgrade reports correspond with the columns you see in Application Designer's upgrade definition window. Moving from left to right, you see the Name of the object, then other key columns that vary by object type, then the Source and Target status, the Action value and Upgrade flag (*Yes* or *No*).

After these columns are three more that are not included in Application Designer. The first is Attribute. This tells you the type of difference that was found between the two objects. For example, record field Attribute values include *Use/Edit*, which identifies key or audit differences, and Default Field Name (*Def. Fldnm*), which identifies differences in a default value. Lastly, there is a Source column and a Target column. These wide columns display the actual differences between the object definitions. For example, on a *Use/Edit* attribute recfield difference, the Source column might contain *Xlat Table Edit* while the Target column is empty. This means the Source record field has a translate table edit while the Target record field does not.

If you are unsure of the meaning of any value in the last three report columns, open the PeopleTool that edits the particular object. The values in these columns correspond directly to dialog options in the tool.

## Task U-2-2: Using Reports

Oracle delivers several cross-reference reports that you can run to provide information about the inter-relationships between various objects. Oracle delivers these reports in the form of SQRs (found in PS\_HOME\SQR), Crystal Reports (found in PS\_HOME\CRW\ENG), and Queries.

The cross-reference reports include:

Object Type(s)	Report Name	Report Description
Applications and Fields	XRFAPFL	Lists all application windows, such as General Tables, in alphabetical order, as well as the fields within each window. For each field, the report details the Field Name, Field Type, Length, and Format, as well as all the record and page definitions that contain the field (within the window).
Fields Referenced by PeopleCode Programs	XRFFLPC	Lists all PeopleCode programs in alphabetical order by associated record definition/field. The report includes type of field and lists all fields referenced in the PeopleCode program.
Fields and Panels	XRFFLPN	Lists all fields in alphabetical order. The report includes the names of all record and page definitions in which each field is used, as well as the Long Name of each field.

Object Type(s)	Report Name	Report Description
Records and Fields	XRFFLRC	Lists all fields in alphabetical order. The report details the Long Name, Field Type, Field Length, and Formatting specified for the field, and includes the names of all record definitions that contain the field.
Field Listing	XRFIELDS	Lists all fields in alphabetical order. The report includes Field Type, Length, Format, Long Name and Short Name.
Menu Listing	XRFMENU	Lists application windows in alphabetical order. The report details all menus within each window, and all page definitions within each menu. It also includes the associated search record definition name and detail page definition name.
Panel Listing	XRFPANEL	Lists all page definitions in alphabetical order.
PeopleCode Programs and Field References	XRFPCL	Lists record definitions that contain fields with PeopleCode program attributes. The report includes the Field Name, as well as the associated record definitions and fields referenced in the PeopleCode program.
Panels with PeopleCode	XRFPNPC	Lists all pages that contain fields with PeopleCode attributes. For each page, the report includes the name of the record definition(s) that contain the field as well as the Field Name and Type.
Fields and Records	XRFRCL	Lists all fields in alphabetical order by associated record definition name. The report details the Long Name, Field Type, Field Length, and Formatting specified for the field.
Records and Panels	XRFRCPN	Lists all record definitions in alphabetical order. The report includes the menu and page definitions associated with each record definition.
Window Listing	XRFWIN	Lists all application windows in alphabetical order.

In addition to using our standard cross-reference reports, you can also generate ad hoc reports to extract the exact combination of information you need. Or, you can create permanent custom reports for information you extract on a regular basis.

Oracle recommends that you mark your upgrade reports using a color-coding system to help you quickly identify what you need to do to certain objects.

If you have several people reviewing sections of the reports, a good documentation policy is to have everyone on your review cycle initial and date the action defaults and overrides they select.

You may also find it easier to change some objects manually after the upgrade, rather than copying the new versions from the source.

### **See Also**

*Enterprise PeopleTools PeopleBook: Crystal Reports for PeopleSoft*





## APPENDIX V

# Using the Data Archiving Tool

---

## Data Archiving Tool

This appendix contains information for customers who have the Rapid Apps (or Strategic Customization Service) Data Archiving Tool. The following tables need to be dropped from your database prior to running the task Updating PeopleTools System Tables. If you do not drop these tables prior to running this task, you will receive errors because the upgrade process will try to create these tables.

- PS\_ARCH\_AUDIT
- PS\_ARCH\_COMMON\_KEY
- PS\_ARCH\_CTRL
- PS\_ARCH\_FLT\_OPT
- PS\_ARCH\_FLT\_RQST
- PS\_ARCH\_HST\_OPT
- PS\_ARCH\_HST\_RQST
- PS\_ARCH\_IMP\_OPT
- PS\_ARCH\_IMP\_RQST
- PS\_ARCH\_LNG\_TMP
- PS\_ARCH\_OTH\_CTRL
- PS\_ARCH\_PROJ
- PS\_ARCH\_RPT\_RQST
- PS\_ARCH\_RQST
- PS\_ARCH\_RQST\_OPT
- PS\_ARCH\_SECURITY
- PS\_ARCH\_SQL\_LNG
- PS\_ARCH\_SQRFLD
- PS\_ARCH\_TBL

Dropping these tables results in the loss of your archive templates. Archive templates created from the Rapid Apps Data Archive Tool cannot be upgraded and are not compatible with archive templates used by the PeopleTools 8.4x Data Archive Manager. If you want to use the templates you previously defined in the Rapid Apps Data Archive Tool, redefine them in the Data Archive Manager.



# Index

## A

- ADP\_ACCT\_CD\_VW view,
  - dropping 137
- AE SYNCIDGEN, running 190
- ALLTABS project, creating 164
- ALLTEMPTABS project, creating 163
- Alter Audit
  - final, reviewing 212
  - final, running 211
  - running initial 11
- alter tables script, running 166
- alter with deletes script
  - building 189
  - running 189
- alternative overtime upgrade,
  - validating 238
- Alternative Overtime, setting values 32
- application audits, running 16
- application changes, preparing 135
- application conversion data,
  - exporting 180
- application conversion data,
  - importing 180
- Application Engine driver program,
  - using 433
- application messages
  - exporting 175
  - importing 175
- application messaging
  - objects, deleting 116
  - objects, saving 115
- application processes, completing 205
- application servers
  - configuring 182, 216
  - granting administrator permissions 68
- application system data
  - exporting 195
  - importing 195
- application tablespaces, creating 100
- application upgrade, preparing 59
- applying
  - fixes after copying project 280
  - fixes after data conversion 281
  - fixes between upgrade passes 281
  - fixes during installation 280

- fixes during Move to Production 282
  - updates before data conversion 181
- UPGOPT project 6
- upgrade planning files 5
- Approval Framework definitions
  - exporting 198
  - importing 198
- approval rule sets
  - altering 110
  - modifying related tables 291
- approvals, preparing for upgrade 25
- assigning upgrade default values 28
- auditing
  - character length semantics 132
  - disabled constraints 125
  - duplicate length constraints 125
  - long to LOB conversion 130
- audits
  - Alter Audit 11
  - ChartFields, running 24
  - DDDAUDIT, final 210
  - DDDAUDIT, initial 10
  - final 210
  - initial 10
  - SYSAUDIT, final 211
  - SYSAUDIT, initial 11

## B

- backing up
  - after data conversion 188
  - after PeopleTools upgrade 133
  - after preparing your database 65
  - after upgrade copy 160
  - before conversion 183
  - before manual changes 223
  - before platform changes 121
  - before testing 269
  - Copy of Current Demo database 51
  - demo databases 51
  - new release demo again 156
  - New Release Demo database 51
- base data, loading 106
- benefit program, understanding 316
- benefit rates, reviewing consolidation 319

- benefit record number display,
  - reviewing 315
- benefits
  - preparing benefits administration for
    - upgrade 26
  - preparing COBRA administration for
    - upgrade 27
  - preparing for upgrade 26
- Benefits Limit Table, reviewing 236
- building
  - alter with deletes script 189
  - PPLTOOLS project 102
  - tablespace alter script 105
  - UPGOPT project 6

## C

- Campus Community, assigning
  - defaults 35
- Change Assistant
  - creating new job 273
  - editing multilingual step properties 78
- change control
  - reviewing 268
  - turning off 89
- character length script 1, running 130
- character length script 2, running 130
- character length script 3, running 130
- character length script 4, running 131
- character length script 5, running 131
- character length script 6, running 131
- character length script 7, running 131
- character length script 8, running 132
- character length semantics, auditing 132
- ChartFields audit reports, running 24
- cloning US garnishment rules 258
- COBOL
  - setting PSOPTIONS for 200
- COBRA administration, preparing for
  - upgrade 27
- common portal system options
  - exporting 196
  - importing 197
- company cars 316
- company cars, effective dating 316
- compare options, modifying 144
- compare reports
  - columns 441
  - reviewing 147
  - understanding 440
  - using 441
- comparing
  - customizations 54
  - fields in PPLTOOLS project 92
  - new release 143
  - PeopleTools objects 92
  - running UPGCUST 55
- comparison process
  - default actions for objects 439
  - settings for objects 440
  - status of objects 438
  - Upgrade column 439
- completing system setup
  - exporting data 192–193
  - exporting EDI statements 192
  - exporting mass change data 193
  - importing data 192, 194
  - importing EDI statements 192
  - importing mass change data 193
  - loading data 191
- configuring
  - application servers 182, 216
  - Portal 216
  - Process Scheduler 182
  - upgrade environment 216
- CONNECT ID, granting permissions 123
- connect ID, granting privileges to 83
- contacting global support 289
- Content Provider registry
  - upgrading 266
- content reference permissions,
  - synchronizing 220
- control tables, populating 181
- conversion
  - loading data 113
  - reporting details 113
  - running data conversion 113
- conversion reports, reviewing 132
- conversion scripts
  - Oracle, generating 126
- converting
  - database data types 119
  - Integration Broker 114
  - PeopleTools objects 109
  - Portal objects 111
  - query prompt headings 112
- Copy of Current Demo database
  - backing up 51
  - restoring 56
- Copy of Production
  - creating 5

- testing 269
- Copy of Production database
  - backing up 133
- Copy of Production database cache,
  - deleting 418
- copy results, reviewing 150
- copying
  - Enterprise Portal database 417
  - PATCH84X project 98
  - PATCH84XML project 99
  - PPLDELETE project 96
  - PPLTLS84CUR project 96
  - PPLTLS84CURDEL project 98
  - PPLTLS84CURML project 97
  - PPLTLSML project 97
  - PPLTOOLS project 95
  - production database 5
  - projects 94
  - PT84TBLSPC project 105
  - PTDDLUPG script 70
  - PTPORTALDEL project 95
  - Time and Labor temp table list 60
  - UPGCUST 150
  - UPGCUSTIB 152
  - UPGIBCOPY 154
  - UPGNONCOMP 154
- coverage calculations, reviewing
  - consolidation 320
- create and alter scripts
  - building 164
  - editing 165
- create indexes script, running 190
- create tables script
  - running 166
- create temp tables script
  - building 163
  - running 165
- create triggers script, running 190
- create upgrade tables script
  - building 162
- CREATEVW, running 190
- creating
  - all views 190
  - ALLTABS project 164
  - ALLTEMPTABS project 163
  - application tablespaces 100
  - copy of RecField definitions 59
  - custom tablespaces 159
  - FNLALTAUD project 211
  - indexes 103, 166

- Integration Broker objects 114
- Microsoft conversion project 123
- new Change Assistant job 273
- new tablespaces 158
- Oracle audit tables 124
- Oracle-delivered tablespaces 158
- PPLTOOLS views 109
- RUNSTATS.DAT 87
- tablespaces 82
- tablespaces for Informix 82
- target Enterprise Portal project 419
- temporary performance indexes 86
- triggers 103
- updated PeopleTools views 109
- UPGIBCOPY 153
- US custom garnishment rules 258
- customizations
  - comparing 54
  - identifying 57
  - reapplying 217
- customized objects
  - adjusting 218
- customizing
  - new release 150
  - template built rules 245

## D

- data
  - base data, loading 106
  - conversion data, loading 113
  - data conversion, running 113
  - English messages, loading 108
  - English string data, loading 108
  - language data, loading 107
  - PeopleTools data, loading 107
  - stored statements data, loading 108
- data conversion
  - applying updates before 181
  - backing up after 188
  - creating indexes 168
  - dropping indexes 167
  - loading data 173
  - loading data for PeopleTools 113
  - reporting details 113
  - running for application changes 183
  - running for group 1 186
  - running for group 10 188
  - running for group 11 188
  - running for group 2 186
  - running for group 20 205

- running for group 3 186
    - running for group 4 187
    - running for group 6 187
    - running for group 7 187
    - running for group 8 187
    - running for PeopleTools changes 113
    - understanding 184
  - data conversion driver data
    - exporting 180
    - importing 180
  - data conversion report 436
  - data model definitions, loading
    - DB2 UNIX/NT 90
    - DB2 z/OS 89
    - Informix 90
    - Oracle 90
    - SQL Server 90
    - Sybase 91
    - understanding 89
  - data type steps, editing 80
  - database
    - increasing space 5
  - database cache
    - deleting from Copy of Production 418
  - database data types, converting 119
  - database options flag, resetting 87
  - database options, updating 133
  - database servers, backing up 133
  - database structure
    - finalizing 188
    - modifying 160
  - databases
    - preparing for the upgrade 48
    - stamping 267
    - updating overrides 156
    - verifying integrity 49
  - DB2
    - editing upgrade planning scripts 8
  - DB2 UNIX
    - generating final RUNSTATS 203
    - running final statistics 204
  - DB2 UNIX RUNSTATS script,
    - generating 172
  - DB2 UNIX/NT
    - loading data model definitions 90
    - rerunning RUNSTATS 87
    - updating statistics 88
  - DB2 z/OS
    - backing up database 161
    - editing create table scripts 74
    - editing MTP import scripts 75
    - loading data model definitions 89
    - running final statistics 204
    - updating statistics 87
  - DB2TMPIDXCREATE script 86
  - DBTSFIX output scripts
    - editing 71
    - running 83
  - DBTSFIX script, running 71
  - DDDAUDIT script
    - final, running 210
    - running initial 10
  - DDLDB2 script
    - editing 76
    - running 89
  - DDLDBX script
    - editing 76
    - running 90
  - DDLIFX script
    - editing 77
    - running 90
  - DDLMS script, running 90
  - DDLORA script
    - editing 77
    - running 90
  - DDLSYB script, running 91
  - default values, assigning 28
  - defining, testing strategy 296
  - deleting
    - application messaging objects 116
    - Copy of Production database cache 418
    - node transactions 116
    - resume files 249
    - unused language pages 118
  - department security, updating 222
  - dependent information, effective
    - dating 316
  - developing
    - test plan 299
    - test scripts 300
  - disabled constraints, auditing 125
  - drop table script, running 135
  - dropping
    - ADP\_ACCT\_CD\_VW view 137
    - student financials tables 60
  - duplicate length constraints, auditing 125
- E**
- eCompensation, completing
    - transactions 40

- ECPrep process, running 233
- eDevelopment, completing transactions 41
- EDI tables, identifying 57
- editing
  - data type steps 80
  - DB2 z/OS create table scripts 74
  - DB2 z/OS MTP import scripts 75
  - DBTSFIX output scripts 71
  - DDLDB2 script 76
  - DDLDBX script 76
  - DDLIFX script 77
  - DDLORA script 77
  - GRANT script 72
  - Integration Broker script 78
  - Move to Production import scripts 75
  - MSGTLSUPG script 77
  - multilingual step properties 78
  - MVPRDIMP script 76
  - PeopleTools script 102
  - PORTALNODES script 79
  - PPLTLS84CURTABLES script 104
  - PSLANG script 72
  - PSLANGUAGES script 73
  - PTDDLUPG script 70
  - PTxxxTLS scripts 73
  - RELGA script 72
  - system data scripts 195
  - tablespace alter script 105
  - TLSUPGNONCOMP script 73
  - upgrade planning DB2 scripts 8
- EE Garn Payee Data Validation Report, running 253
- effective dating 316
  - dependent information 316
- employee ID length, updating 264
- Enterprise Portal
  - copying database 417
  - upgrading PeopleTools 417
- Enterprise Portal project, target creating 419
- eProfile approvals, completing 26
- equation conversion reports, running 230
- equation conversion, verifying 230
- equation security, reviewing 232
- equation values
  - mapping 41
- equations, reviewing 231
- expanding fields on records 110

- Expenses ChartField mappings, setting up 252
- exporting
  - application conversion data 180
  - application messages 175
  - application system data 195
  - Approval Framework definitions 198
  - common portal system options 196
  - data conversion driver data 180
  - delivered equations 179
  - generic notifications 198
  - GL interface setup tables 207
  - new release objects 155
  - node transactions 115
  - Pagelet Wizard application data 177
  - Pagelet Wizard data 178
  - PATCH84X project 98
  - PATCH84XML project 99
  - payroll interface tables 206
  - PeopleTools system tables 86
  - PPLDELETE project 96
  - PPLTLS84CUR project 96
  - PPLTLS84CURDEL project 98
  - PPLTLS84CURML project 97
  - PPLTLSML project 97
  - PPLTOOLS project 95
  - project definitions 136
  - projects 94
  - PT84TBLSPC project 105
  - PTPORTALDEL project 95
  - record groups 175
  - related language system data 194
  - rules setup data 206
  - selected PeopleTools tables 151
  - setup data 197
  - system data for upgrade path 196
  - system setup data 176
  - upgrade default options 43
  - upgrade defaults 178
  - US custom garnishment rules 208

## F

- field compare report, reviewing 94
- fields
  - renaming 51
- fields for Microsoft SQL Server 138
- file servers
  - editing PTxxxTLS scripts 73
- final audit reports, running 210
- Financial Aid equations, updating 42

- fixes
  - applying after copying project 280
  - applying after data conversion 281
  - applying between upgrade passes 281
  - applying during installation 280
  - applying during Move to Production 282
- FNLALTAUD project, creating 211
- FSA balances, reviewing consolidation 313
- functional decisions, making 27

**G**

- garnishment payee data
  - updating 255
  - validating 253
- garnishment payee data, correcting 21
- garnishment rule data
  - running report 260
  - updating 261
  - validating 259
- garnishment rules
  - exporting 208
  - importing 209
  - US custom, creating 258
  - US custom, setting up 257
  - US, cloning 258
- garnishment rules changes, understanding 373
- garnishments upgrade, understanding 371
- General Ledger activity groupings, setting up 252
- General Ledger interface, setting up 249
- generating
  - DB2 UNIX RUNSTATS script 172
  - final RUNSTATS 203
  - Microsoft conversion scripts 123
  - Oracle conversion scripts 126
  - PeopleTools script 102
  - PPLTLS84CURTABLES script 103
- generic notifications
  - exporting 198
  - importing 199
- getting started 1
- GL interface setup tables
  - exporting 207
  - importing 208
- global support, contacting 289
- GRANT script
  - editing 72

- running 83
- granting
  - home page personalization access 221
  - privileges to connect ID 83
- granting permissions
  - CONNECT ID 123

**H**

- home page personalization access, granting 221
- HR 8.01 default values, assigning 28
- HR default values, assigning 37
- HR ethnicities, loading 136

**I**

- identifying
  - customizations 57
  - EDI tables 57
  - mass change 57
- images, shrinking 49
- importing
  - application conversion data 180
  - application messages 175
  - application system data 195
  - Approval Framework definitions 198
  - common portal system options 197
  - data conversion driver data 180
  - DB2 z/OS-specific information 75
  - delivered equations 179
  - generic notifications 199
  - GL interface setup tables 208
  - new release objects 155
  - Pagelet Wizard application data 177
  - Pagelet Wizard data 178
  - payroll interface tables 207
  - PeopleTools system tables 86
  - project definitions 136
  - record groups 175
  - related language system data 194
  - rules setup data 206
  - selected PeopleTools tables 152
  - setup data 198
  - system data for upgrade path 196
  - system setup data 177
  - upgrade default options 9
  - upgrade defaults 179
  - US custom garnishment rules 209
- increasing space, log file and database 5
- index parameters



- setting 171
- indexes
  - creating 103, 166
  - dropping for data conversion 167
  - parameters, setting after copy 156
  - reviewing the create indexes log 167
  - temporary performance, creating 86
- Informix
  - loading data model definitions 90
  - running final statistics 204
  - updating message nodes for 111
  - updating statistics 88
- initial audits
  - reviewing 15
  - running 10
- installation, applying fixes during 280
- Integration Broker
  - converting 114
  - defaults, updating 114
  - deletes, preparing 115
  - objects, creating 114
- Integration Broker script, editing 78
- integration testing 297

## L

- language data, updating 209
- language system data
  - exporting related 194
  - importing related 194
- languages
  - deleting unused pages 118
  - loading data 107
  - populating 107
  - populating for login 84
  - swapping on system data 174
  - verifying nonbase pages 116
- liability accounts, setting up 250
- license code, updating 84
- loading
  - base data 106
  - conversion data 113
  - data for data conversion 173
  - data model definitions 89
  - data to complete system setup 191
  - English messages 108
  - English string data 108
  - HR ethnicities 136
  - language data 107
  - message data 91
  - noncomparable objects 108

- PeopleTools data 107
- planning messages and strings 9
- stored statements 199
- stored statements data 108
- system messages 91
- local message node, preserving 144
- log file, increasing space 5
- login, populating languages for 84
- long data audit, running 122
- long to LOB conversion, auditing 130
- long to LOB script 1, running 128
- long to LOB script 2, running 128
- long to LOB script 3, running 128
- long to LOB script 4, running 128
- long to LOB script 5, running 129
- long to LOB script 6, running 129
- long to LOB script 7, running 129
- long to LOB script 8, running 129

## M

- manual conversion candidates,
  - reviewing 231
- mapping
  - equation values 41
- mapping options 30
- mass change, identifying 57
- message data
  - loading 91
  - loading English messages 108
- message data, cleaning up 81
- message nodes
  - script, editing 79
  - updating 110
  - updating for Informix 111
- message queues, purging 50
- Microsoft conversion project,
  - creating 123
- Microsoft conversion report, running 124
- Microsoft conversion scripts,
  - generating 123
- Microsoft conversion scripts, running 123
- Microsoft database, validating 122
- Microsoft settings, reviewing 122
- Microsoft SQL Server, renaming fields
  - for 138
- Microsoft SQL Server, reversing renames
  - for 61
- migrating records 104
- model definitions, *See* data model definitions

- modifying
  - approval rule set-related tables 291
  - compare options 144
  - database structure 160
  - SQL statements for PROCESS\_INSTANCE 248
  - user exit rules 247
- Move to Production
  - editing import scripts 75
  - editing password 76
  - performing 274
  - testing 271
  - understanding 272
- MSGTLSUPG script, editing 77
- multilingual step properties, editing 78
- MVPRDIMP script, editing 76

## N

- New Copy of Production database
  - importing data 75
- new release
  - exporting objects 155
  - importing objects 155
- new release database
  - backing up again 156
- new release demo
  - restoring 181
- New Release Demo database
  - backing up 51
- node transactions
  - deleting 116
  - exporting 115
- non-comparable objects, reviewing 148
- nonbase language pages, verifying 116
- notes and tips, for your upgrade 3

## O

- object version numbers
  - setting 119
  - updating 210
- object version numbers, resetting 155
- Operator Security Join Table,
  - refreshing 222
- Oracle
  - loading data model definitions 90
  - running final statistics 205
  - updating statistics 88
- Oracle audit tables, creating 124
- Oracle conversion scripts, generating 126

- Oracle database, validating 124
- Oracle settings, reviewing 125

## P

- Pagelet Wizard
  - exporting application data 177
  - exporting data 178
  - importing application data 177
  - importing data 178
- parallel testing 298
- password, Move to Production 76
- patch information, updating 85
- PATCH84X project 98
- PATCH84XML project 99
- payee changes, understanding 371
- Payroll Interface Definitions,
  - updating 238
- payroll interface tables
  - exporting 206
  - importing 207
- pension, setting up 235
- PeopleSoft upgrade survey 274
- PeopleTools
  - comparing objects 92
  - data, loading 107
  - editing script 102
  - exporting system tables 86
  - functionality 265
  - generating script 102
  - importing system tables 86
  - objects, converting 109
  - re-exporting tables 308
  - renaming tables 91
  - running script 103
  - script 103
  - swap script, editing 64
  - updating patch information 85
  - updating system tables 80
  - upgrade, backing up after 133
  - upgrading Enterprise Portal 417
- PeopleTools tables
  - exporting 151
  - importing 152
  - swapping languages 152
- performance
  - improving for your platform 288
  - testing 298
  - tips for creating and altering SQL 288
  - tips for data conversion 288
  - tips for DB2 289

- tips for modify database structures 288
- tips for Oracle RDBMS 288
- Performance Management, assigning defaults 29
- permission lists
  - updating Portal registry 418
- permissions lists
  - understanding registry updates 417
- person model conversion method, defining 37
- person organization audits, running 234
- personal data
  - running setup script 137
- personal data changes, running report 22
- personal data table, refreshing 205
- PI Field Definition Table, updating 240
- PI Instance Table definitions, updating 241
- planning messages and strings, loading 9
- planning scripts, running 8
- platform changes, backing up before 121
- populating
  - equation renames 9
  - recruiting solutions 9
- Portal
  - configuring 216
  - converting objects 111
  - running security synchronization 220
- Portal navigation objects, registering 218
- portal options data, updating 267
- Portal registry
  - updating permission lists 418
- PORTALNODES script
  - editing 79
- PPLDELETE project 96
- PPLTLS84CUR project 96
- PPLTLS84CURDEL project 98
- PPLTLS84CURML project 97
- PPLTLS84CURTABLES script
  - editing 104
  - generating 103
  - running 104
- PPLTLSML project 97
- PPLTOOLS project
  - building 102
  - comparing fields 92
  - exporting and copying 95
- preparing
  - application upgrade 59
  - for the upgrade 48
- Integration Broker deletes 115
- preserving
  - queries and trees 305
- preserving, local message node 144
- process and report tables, clearing 118
- Process Scheduler
  - configuring 182
- product license code, updating 84
- production database
  - identifying empty tables 48
- production database, copying 5
- Profile Management, assigning options 36
- project definitions
  - exporting 136
  - importing 136
- projects
  - exporting and copying 94
  - PATCH84X 98
  - PATCH84XML 99
  - PPLDELETE 96
  - PPLTLS84CUR 96
  - PPLTLS84CURDEL 98
  - PPLTLS84CURML 97
  - PPLTLSML 97
  - PPLTOOLS 102
  - PT84TBLSPC 105
  - PTPORTALDEL 95
- PS Table definitions, updating 240
- PSLANG script
  - editing 72
  - running 84
- PSLANGUAGES script
  - editing 73
  - running 107
- PSOBJCHNG table, cleaning 49
- PSOPTIONS
  - setting for COBOL 200
- PSX\_PAGEACTION table 116
- PT84RLPD script 118
- PT84TBLSPC project 105
- PTDDLUPG script
  - copying 70
  - editing 70
  - running 82
- PTPORTALDEL project 95
- PTUPGCONVERT program 113
- PTxxxTLS scripts
  - editing 73
  - running 106

**Q**

- queries, preserving 305
- query prompt headings, converting 112

**R**

- re-creating
  - triggers 167
- reapplying customizations 217
- RecField definitions, creating a copy 59
- recompiling, template built rules 242
- record groups
  - exporting 175
  - importing 175
- record parameters
  - setting 172
- records
  - migrating 104
  - renaming 51
  - setting parameters after copy 157
- Recruiting Solutions 30
- registering Portal navigation objects 218
- registry permission lists, understanding updates 417
- registry, upgrading Content Provider 266
- regression testing 299
- relationship usage, reviewing
  - changes 317
- RELGA script
  - editing 72
  - running 84
- renamed tables script
  - building 162
  - running 163
- renames, reversing for Microsoft SQL Server 61
- renaming 138
  - DI tables for DB2 UNIX NT 63
  - DI tables for DB2 z/OS 63
  - DI tables for Informix 62
  - DI tables for Microsoft 61
  - DI tables for Oracle 62
  - DI tables for Sybase 63
  - fields 51
  - PeopleTools tables 91
  - records 51
  - table with data type changes for Microsoft 61
  - tables 138
- reporting conversion details 113
- reports
  - personal data changes 22
  - upgrade, running 16
  - upgrade, understanding 17
- resetting
  - action flags in UPGCUST 146
  - database options flag 87
  - object version numbers 155
- restoring
  - Copy of Current Demo database 56
  - new release demo 181
- resume files, deleting 249
- retaining target rename log files 53
- Retiree Additional Pay Report
  - running UVPYT05.SQR 227
  - running UVPYT06.SQR 227
  - running UVPYT07.SQR 228
- retirees with pay
  - converting 224
  - rebuilding job flags 229
  - running Retiree Additional Pay Report UVPYT05.SQR 227
  - running Retiree Additional Pay Report UVPYT06.SQR 227
  - running Retiree Additional Pay Report UVPYT07.SQR 228
  - running upgrade conversion 228
  - understanding conversion 224
- reviewing 314
  - Alter Audit, final 212
  - Benefits Limit Table 236
  - change control 268
  - conversion reports 132
  - copy results 150
  - create indexes log 167
  - equations 231
  - field compare report 94
  - FSA balances consolidation 313
  - initial audits 15
  - manual conversion candidates 231
  - Microsoft settings 122
  - new release changes 147
  - new release compare reports 147
  - non-comparable objects 148
  - Oracle settings 125
  - PeopleTools functionality 265
  - UPGCUST compare log 56
  - UPGCUSTIB copy results 155
  - UPGIBCOPY copy results 155
  - UPGNONCOMP copy results 155

- user exit rules 247
- RNCSUPG03 script
  - running 53
  - running on Copy of Current Demo 54
- RNHCEOG20 script
  - running 52
  - running on Copy of Current Demo 54
- RNHCUPG01 script
  - running 52
  - running on Copy of Current Demo 53
- rules setup data
  - exporting 206
  - importing 206
- rules, upgrading 241
- running
  - Alter Audit, final 211
  - Alter Audit, initial 11
  - alter tables script 166
  - alter with deletes script 189
  - application audits 16
  - audits 210
  - character length script 1 130
  - character length script 2 130
  - character length script 3 130
  - character length script 4 131
  - character length script 5 131
  - character length script 6 131
  - character length script 7 131
  - character length script 8 132
  - create indexes script 166
  - create tables script 166
  - create temp tables script 165
  - create triggers script 190
  - data conversion, for application
    - changes 183
  - data conversion, for PeopleTools
    - changes 113
  - DB2TMPIDXCREATE script 86
  - DBTSFIX output scripts 83
  - DBTSFIX script 71
  - DDDAUDIT script, final 210
  - DDDAUDIT script, initial 10
  - DDLDB2 script 89
  - DDLDBX script 90
  - DDLIFX script 90
  - DDLMS script 90
  - DDLORA script 90
  - DDLSYB script 91
  - DLHCCOG03E script 197
  - DLHCCOG03I script 197
  - dropped temp tables report 241
  - ECPrep process 233
  - EE Garn Payee Data Validation Report 253
  - equation conversion reports 230
  - final statistics for DB2 UNIX 204
  - final statistics for DB2 z/OS 204
  - final statistics for Informix 204
  - final statistics for Oracle 205
  - final update statistics 203
  - garnishment rule data report 260
  - GRANT script 83
  - initial audits 10
  - long data audit 122
  - long to LOB script 1 128
  - long to LOB script 2 128
  - long to LOB script 3 128
  - long to LOB script 4 128
  - long to LOB script 5 129
  - long to LOB script 6 129
  - long to LOB script 7 129
  - long to LOB script 8 129
  - Microsoft conversion report 124
  - Microsoft conversion scripts 123
  - new release UPGCUST 146
  - new release UPGCUSTIB 146
  - new release upgrade copy 151
  - PeopleTools script 103
  - person organization audits 234
  - personal data setup script 137
  - Portal security synchronization 220
  - PPLTLS84CURTABLES script 104
  - PSLANG script 84
  - PSLANGUAGES script 107
  - PT84RLPD script 118
  - PTDDLUPG script 82
  - PTUPGCONVERT program 113
  - PTxxxTLS scripts 106
  - RELGA script 84
  - RNCSUPG03 script 53
  - RNHCEOG20 script 52
  - RNHCUPG01 script 52
  - row count report 48
  - SETINDEX script 156, 212
  - SETSPACE script 157
  - SETTABLE script 157, 213
  - STOREAD script 201
  - STOREAV script 201
  - STOREBAS script 202
  - STORECC script 200

- STOREFA script 200
- STOREGP script 202
- STOREHRM script 202
- STOREPAY script 202
- STOREPEN script 203
- STOREPYI script 203
- STORESF script 201
- STORESR script 201
- SYSAUDIT script, final 211
- SYSAUDIT script, initial 11
- tablespace alter script 106
- TLSUPGNONCOMP script 108
- TSRECPOP script 209
- UPGCOUNT script 48
- UPGCUST 55
- UPGCUST filter script 56
- upgrade planning scripts 8
- upgrade reports 16
- XLATTABLE copy script 59
- RUNSTATS.DAT, creating 87

## S

- saving
  - application messaging objects 115
- savings management 314
- savings management, reviewing 314
- schedules, assigning defaults 33
- scripts
  - DB2 z/OS create table scripts 74
  - DB2 z/OS MTP import scripts 75
  - DB2TMPIDXCREATE 86
  - DBTSFIX, editing 71
  - DBTSFIX, running 71
  - DDDAUDIT 10
  - DDLDB2, editing 76
  - DDLDB2, running 89
  - DDLDBX, editing 76
  - DDLDBX, running 90
  - DDLIFX, editing 77
  - DDLIFX, running 90
  - DDL MSS, running 90
  - DDLORA, editing 77
  - DDLORA, running 90
  - DDLSYB, running 91
  - DHC8X01DMS 175
  - DLHCCOG03E 197
  - DLHCCOG03I 197
  - editing DB2 upgrade planning 8
  - GRANT 83
  - GRANT, editing 72

- Integration Broker 78
- message nodes 79
- MSGTLSUPG 77
- MVPRDIMP 76
- PeopleTools 103
- PORTALNODES 79
- PPLTLS84CURTABLES 103–104
- PSLANG 72, 84
- PSLANGUAGES 107
- PSLANGUAGES, editing 73
- PT84RLPD 118
- PTDDLUPG 70
- PTxxxTLS scripts 106
- PTxxxTLS scripts, editing 73
- RELGA, editing 72
- RELGA, running 84
- running ALLTABS\_ALTTLBL 166
- running ALLTABS\_CRTTLBL 166
- running RNCSUPG02DB2 142
- running RNCSUPG02DBX 142
- running RNCSUPG02IFX 143
- running RNCSUPG02MSS 142
- running RNCSUPG02ORA 143
- running RNCSUPG02SYB 143
- running RNCSUPR04DB2 141
- running RNCSUPR04DBX 141
- running RNCSUPR04IFX 141
- running RNCSUPR04MSS 140
- running RNCSUPR04ORA 141
- running RNCSUPR04SYB 142
- running RNHCUPG03DB2 139
- running RNHCUPG03DBX 139
- running RNHCUPG03IFX 140
- running RNHCUPG03MSS 139
- running RNHCUPG03ORA 140
- running RNHCUPG03SYB 140
- SETINDEX 156
- SETSPACE 157
- SETTABLE 157
- SYSAUDIT 11
- TLSUPGNONCOMP 108
- TLSUPGNONCOMP, editing 73
- UPGCOUNT 48
- security
  - granting home page personalization access 221
  - running Portal security synchronization 220
  - setting up 219
  - setting up, upgrade planning 7

- synchronizing content reference
  - permissions 220
  - understanding setup 219
- Security Join Tables, refreshing 223
- servers
  - application servers 182, 216
  - database servers 133
  - file servers 73
  - web servers 79
- Service Indicators, upgrading defaults 44
- SETINDEX script 156, 212
- SETSPACE script 157
- SETTABLE script 157, 213
- setting 119
  - index parameters 156
  - object version numbers 119
  - PSOPTIONS for COBOL 200
  - record parameters after copy 157
  - tablespace names 157
- setting up
  - pension 235
  - security 219
  - security for upgrade planning 7
  - Stock Administration 236
  - stock purchase plan rules 237
  - Stock Table 237
  - student group security 265
  - US custom garnishment rules 257
- settings
  - Microsoft, reviewing 122
  - Oracle, reviewing 125
- setup data
  - exporting 197
  - importing 198
- shrinking images 49
- single signon, setting up 216
- snapshot tables, changes to 314
- software installation
  - verifying 2
- spending account changes, reviewing 317
- SQL Server
  - loading data model definitions 90
- SQL statements
  - modifying for PROCESS\_INSTANCE 248
- stamping the database 267
- starting
  - application servers 216
- statistics
  - DB2 UNIX/NT, updating 88
  - DB2 z/OS, updating 87
  - Informix, updating 88
  - Oracle, updating 88
  - running final 203
  - updating 10
- Stock Administration
  - setting up 236
- stock purchase plan rules, setting up 237
- Stock Table, setting up 237
- STOREAD script, running 201
- STOREAV script, running 201
- STOREBAS script, running 202
- STORECC script, running 200
- stored statements data, loading 108
- stored statements, loading 199
- STOREFA script, running 200
- STOREGP script, running 202
- STOREHRM script, running 202
- STOREPAY script, running 202
- STOREPEN script, running 203
- STOREPYI script, running 203
- STORESF script, running 201
- STORESR script, running 201
- string data, loading 108
- Student Financials equations, updating 43
- student financials tables, dropping 60
- Student Financials, assigning defaults 38
- student group security, setting up 265
- Student Records, assigning defaults 38
- support, contacting 289
- Sybase
  - loading data model definitions 91
- synchronizing content reference
  - permissions 220
- SYSAUDIT script
  - running final 211
  - running initial 11
- system
  - messages, loading 91
  - tables, exporting 86
  - tables, importing 86
- system catalog views, updating 82
- system data
  - exporting for upgrade path 196
  - importing for upgrade path 196
- system data scripts, editing 195
- system data swap script, editing 64
- system setup data
  - exporting 176
  - importing 177

system testing 297

## T

### tables

- creating and altering 103
- moving to new tablespaces 106
- PeopleTools system tables, updating 80
- PeopleTools tables, renaming 91
- process and report tables, clearing 118
- PSOBJCHNG 49
- PSX\_PAGEACTION 116
- renaming 138
- running drop script 135
- running row count report 48
- student financials, dropping 60
- system tables, exporting 86
- system tables, importing 86

### tablespaces

- alter script 106
- alter script, building 105
- alter script, editing 105
- creating 82
- creating custom 159
- creating for Informix 82
- creating new 158
- creating Oracle-delivered 158
- migrating records to 104
- populating data 99
- setting names 157, 171
- setting names for temporary tables 171
- updating names 101

target rename log files, retaining 53

target values, updating 154

### template built rules

- customizing 245
- recompiling 242

temporary tables, setting tablespace names 171

test plan, developing 299

test scripts, developing 300

### testing

- after the upgrade 274
- backing up before 269
- Copy of Production 269
- determining conditions 299
- evaluating requirements 296
- integration 297
- Move to Production 271
- parallel environments 298
- performance 298

regression 299

system 297

tips and techniques 301

unit 297

user acceptance 298

### testing strategy

- defining 296
- determining conditions 299
- when to test 295

### TLSUPGNONCOMP script

- editing 73
- running 108

training employee record defaults, assigning 39

transaction records, refreshing 223

### transactions

- eCompensation 40
- eDevelopment 41
- preparing workflow 40

Tree/Query Copy of Production

- comparing project 307
- copying project 308
- creating project in 306
- preparing database 306

trees, preserving 305

triggers, creating 103

triggers, re-creating 167

TSRECPOP script, running 209

turning off change control 89

## U

unit testing 297

### updates

- applying before data conversion 181

### updating

- database options 133
- database overrides 156
- employee ID length 264
- garnishment payee data 255
- garnishment rule data 261
- Integration Broker defaults 114
- language data 209
- message nodes 110
- object version numbers 210
- Payroll Interface Definitions 238
- PeopleTools patch information 85
- PeopleTools system tables 80
- portal options data 267
- product license code 84



- statistics for DB2 UNIX during
    - application changes 172
  - statistics for DB2 UNIX/NT 88
  - statistics for DB2 z/OS 87
  - statistics for DB2 z/OS during application
    - changes 173
  - statistics for Informix 88
  - statistics for Informix during application
    - changes 173
  - statistics for Oracle 88
  - statistics for Oracle during application
    - changes 173
  - statistics, initial 10
  - system catalog views 82
  - tablespace names 101
  - target values 154
  - updating statistics
    - DB2 UNIX/NT 88
    - DB2 z/OS 87
    - Informix 88
    - Oracle 88
  - UPGCOUNT script, running 48
  - UPGCUST
    - copying 150
    - resetting action flags 146
    - reviewing compare log 56
    - running a compare 55
    - running filter script 56
    - running new release 146
  - UPGCUSTIB
    - copying 152
    - reviewing copy results 155
    - running filter script 147
    - running new release 146
  - UPGCUSTIB filter script 147
  - UPGIBCOPY
    - copying 154
    - creating 153
    - reviewing copy results 155
  - UPGNONCOMP
    - copying 154
    - reviewing copy results 155
  - UPGOPT project
    - applying 6
    - building 6
  - upgrade
    - compare reports, understanding 440
    - configuring environment 216
    - database preparation 1
    - databases, defined 2
    - default values, assigning 28
    - getting started 1
    - notes and tips 3
    - PeopleTools, backing up after 133
    - planning for performance 287
    - preparing for 48
    - reports, running 16
    - setting up upgrade planning security 7
    - user, verifying 68
  - upgrade copy, running 151
  - upgrade default options
    - exporting 43
    - importing 9
  - upgrade defaults
    - exporting 178
    - importing 179
  - Upgrade Drivers page
    - accessing 433
    - adding new section 434
    - inactivating section 435
  - upgrade planning files, applying 5
  - upgrade tables script
    - recreating 162
  - upgrading
    - Content Provider registry 266
    - Enterprise Portal PeopleTools 417
    - rules 241
  - US garnishment rules
    - cloning 258
    - creating 258
    - setting up 257
  - user acceptance testing 298
  - user exit rules
    - modifying 247
  - user interface
    - changing for desktop computers 283
    - changing for mobile devices 285
  - using
    - Application Engine Driver
      - program 433
- ## V
- Validate PI Field References Report,
    - running 239
  - validating
    - garnishment payee data 253
    - garnishment rule data 259
    - Microsoft database 122
    - Oracle database 124
  - verifying

- database integrity 49
- nonbase language pages 116
- software installation 2
- upgrade user 68
- version numbers
  - setting 119
  - updating 210
- views
  - creating all 190
  - creating PPLTOOLS 109

## **W**

- web servers
  - editing message nodes script 79
- workflow transactions, preparing 40

## **X**

- XLATTABLE copy script
  - running 59