

**ORACLE®**

PEOPLESOFT

---

**Oracle's PeopleTools PeopleBook**

# PeopleTools 8.52: PeopleSoft Application Designer Developer's Guide

---

**February 2012**

**ORACLE®**

Copyright © 1988, 2012, Oracle and/or its affiliates. All rights reserved.

### **Trademark Notice**

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

### **License Restrictions Warranty/Consequential Damages Disclaimer**

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

### **Warranty Disclaimer**

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

### **Restricted Rights Notice**

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

### **Hazardous Applications Notice**

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

### **Third Party Content, Products, and Services Disclaimer**

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

# Contents

## Preface

<b>Oracle's PeopleSoft Application Designer Preface .....</b>	<b>xxi</b>
Oracle's PeopleSoft Application Designer .....	xxi
PeopleBooks and the PeopleSoft Online Library .....	xxi

## Part 1

### Getting Started with PeopleSoft Application Designer

#### Chapter 1

<b>Getting Started with PeopleSoft Application Designer .....</b>	<b>3</b>
PeopleSoft Application Designer Overview .....	3
PeopleSoft Application Designer Implementation .....	3
Installation-Level Steps .....	3
Main Application Development Steps .....	4

## Part 2

### Developing and Customizing PeopleSoft Applications

#### Chapter 2

<b>Using PeopleSoft Application Designer .....</b>	<b>9</b>
Understanding PeopleSoft Application Designer .....	9
Using PeopleSoft Application Designer .....	9
Understanding the Eight Steps of Application Development .....	10
Starting PeopleSoft Application Designer .....	11
Using the PeopleSoft Application Designer User Interface .....	14
Setting PeopleTools Options .....	22
Using the PeopleCode and Other Programming Language Editors .....	24
Understanding PeopleCode and Other Programming Languages .....	24

Using the Programming Editors .....	25
Working with Definitions .....	27
Understanding Definitions .....	28
Opening Definitions .....	30
Viewing Definitions .....	31
Viewing and Editing Definition Properties .....	31
Creating Definitions .....	33
Saving Definitions .....	33
Closing Definitions .....	34
Renaming Definitions .....	34
Deleting Definitions .....	35
Finding Definitions .....	36
Using the Multiple Document Interface .....	38
Using Drag-and-Drop Mouse Techniques .....	39
Using Dynamic Toolbars and Menus .....	40
Using Pop-up Menus .....	40
Configuring Environments .....	41
Working with Generated Files .....	42
Using Miscellaneous Definitions .....	42
Using the PeopleSoft Application Designer Spell Check Feature .....	44
Understanding the Spell Check Feature .....	44
Enabling the Spell Check Feature .....	48
Configuring Browser Options .....	48
Building and Maintaining Data .....	49
Building and Maintaining Data .....	50
Creating SQL Tables .....	50

## Chapter 3

<b>Working With Projects .....</b>	<b>51</b>
Understanding Projects .....	51
Viewing Projects .....	52
Accessing Project Definitions .....	52
Using the Upgrade View .....	53
Creating and Saving Projects .....	53
Creating New Projects .....	54
Saving Projects .....	54
Setting Project Properties .....	54
Inserting Definitions Into Projects .....	55
Inserting Individual Definitions into a Project .....	56
Inserting a Group of Definitions into a Project .....	56
Removing Definitions From Projects .....	57
Merging Projects .....	57

Using Change Projects .....	59
Setting Project Options .....	59
Validating Projects .....	60
Setting System IDs .....	61

## Chapter 4

<b>Creating Field Definitions .....</b>	<b>63</b>
Understanding Field Definitions .....	63
Creating New Field Definitions .....	63
Prerequisite .....	64
Creating a New Field Definition .....	64
Specifying Attributes for a New Field Definition .....	65
Specifying Character Field Attributes .....	67
Specifying Long Character Field Attributes .....	69
Specifying Number Field Attributes .....	70
Specifying Date Field Attributes .....	72
Specifying Time Field Attributes .....	72
Specifying DateTime Field Attributes .....	72
Specifying Image Field Attributes .....	73
Specifying ImageReference Field Attributes .....	73
Working With Additional Oracle Data Type Mappings .....	74
Working With Additional Microsoft SQL Server 2005 Data Type Mappings .....	75
Using Multiple Labels .....	75
Understanding Multiple Labels .....	75
Adding a New Label .....	76
Deleting a Label .....	77
Selecting Record Field Labels .....	77
Opening an Existing Field Definition .....	78
Viewing Field Properties .....	79
Changing Field Definitions .....	80
Understanding the Effects of Changing Field Definitions .....	80
Changing Field Types .....	80
Renaming Field Definitions .....	81
Deleting Field Definitions .....	82
Printing Field Definitions .....	83
Setting Custom Field Formats .....	83
Understanding Custom Field Formats .....	83
Understanding Format Notation .....	83
Changing Format Families .....	86
Editing Formats .....	88
Using Family Options .....	90
Testing Formats .....	91

Using the Translate Table .....	92
Understanding the Translate Table .....	92
Adding Values to the Translate Table .....	94
Changing Translate Values .....	96
Deleting Translate Values .....	96
Saving the Translate Table .....	96

## Chapter 5

<b>Planning Records, Control Tables, and TableSets .....</b>	<b>97</b>
Understanding the Planning Process .....	97
Table-Based Systems .....	97
Normalized Relational Databases .....	98
Record Definition Planning .....	100
Effective Dates .....	100
Control Tables .....	102
TableSets .....	103
Sharing Tables .....	106
Understanding Table Sharing .....	107
Adding the SetID Field to Record Definitions .....	109
Defining Set Control Fields .....	109
Modifying Set Control Fields .....	110
Creating SetIDs .....	111
Defining Record Groups .....	111
Defining TableSet Controls .....	112
Sharing Trees .....	112

## Chapter 6

<b>Creating Record Definitions .....</b>	<b>115</b>
Understanding Record Definitions .....	115
Understanding Record Types .....	117
Viewing Record Definitions .....	119
Common Elements Used in This Section .....	119
Viewing Basic Field Definitions .....	119
Viewing Key-Related Characteristics and Default Values .....	120
Viewing Editing Options .....	121
Viewing PeopleCode Program Types .....	121
Reordering Fields .....	122
Sizing and Sorting Columns .....	122
Saving Record Definitions .....	123
Naming Record Definitions .....	123

Creating New Record Definitions .....	124
Opening Existing Record Definitions .....	125
Manipulating Fields in Record Definitions .....	125
Understanding Fields in Record Definitions .....	125
Inserting Fields Into Records .....	125
Inserting Subrecords Into Records .....	126
Setting Record Properties .....	127
Setting General Properties .....	127
Setting Use Properties .....	127
Creating User-Defined Audit Record Definitions .....	130
Setting Record Field Properties .....	131
Understanding Record Field Properties .....	131
Understanding Record Key Considerations .....	131
Setting Record Field Use Properties .....	134
Configuring Search Results Grid Columns in Persistent Searches .....	141
Adding From and Through Logic to a Search Page .....	141
Setting Record Field Edit Properties .....	143
Moving Fields in the Same Record .....	146
Moving Fields to Another Record Definition .....	147
Deleting Fields From a Record Definition .....	147
Renaming Record Definitions .....	148
Deleting Record Definitions .....	148
Printing Record Definitions .....	149
Creating SQL View and Dynamic View Select Statements .....	152

## Chapter 7

<b>Administering Data .....</b>	<b>155</b>
Understanding Data Administration and the Build Process .....	155
Running the Build Process .....	156
Understanding the Build Menu .....	157
Selecting the Build Scope .....	159
Selecting Build Options and Running the Build Process .....	159
Using Command Line Syntax to Build a Project .....	161
Creating Tables .....	163
Understanding the Create Table Process .....	163
Creating a Table .....	164
Specifying Create Options .....	165
Specifying Logging Options .....	167
Specifying Script Options .....	171
Confirming a Table Build .....	175
Creating Indexes .....	175
Understanding Indexes .....	176

Configuring Indexes .....	178
Adding a Custom Index .....	179
Editing the DDL .....	180
Altering Tables .....	181
Understanding the Alter Process .....	181
Determining Tables To Be Altered After a Field Change .....	187
Altering a Table .....	188
Specifying Alter Options .....	188
Confirming Your Table Alter .....	192
Creating Triggers .....	192
Creating Views .....	192
Understanding Online Views .....	193
Creating Views .....	193
Using Joins in Views .....	194
Using Cross-Platform Meta-SQL for Dynamic Views .....	194
Administering DDL .....	195
Using the Record DDL .....	195
Setting the Tablespace .....	196
Using Physical Data Storage .....	197

## Chapter 8

<b>Designing Pages for Optimal Performance .....</b>	<b>199</b>
Improving Online Performance .....	199
Understanding Processing Modes .....	199
Refreshing Pages .....	200
Triggering Server Processing .....	200
Using PeopleCode in Deferred Mode .....	201
Using Page Controls in Deferred Mode .....	202
Using Error and Warning Messages .....	205

## Chapter 9

<b>Using Page Controls .....</b>	<b>207</b>
Understanding Page Controls .....	207
Aesthetic Controls .....	208
Data Entry Controls .....	209
Function and Data Processing Controls .....	211
Using Charts .....	212
Using Frames .....	212
Using Group Boxes .....	213
Inserting and Moving a Group Box .....	213



Setting Group Box Properties .....	214
Using Horizontal Rules .....	217
Drawing a Horizontal Rule .....	217
Setting Horizontal Rule Properties .....	217
Using Images .....	218
Understanding Static Images .....	218
Adjusting Static Image Size and Shape .....	219
Setting Static Image Properties .....	219
Setting Image Properties .....	220
Using Static Text .....	221
Using Check Boxes .....	221
Setting Check Box Properties .....	221
Using Drop-Down List Boxes .....	222
Setting Drop-Down List Box Record Properties .....	222
Setting Drop-Down List Box Label and Use Properties .....	223
Using Radio Buttons .....	223
Understanding Radio Buttons .....	223
Setting Radio Button Record Properties .....	224
Setting Radio Button Label Properties .....	224
Setting Radio Button Use Properties .....	224
Using Edit Boxes and Long Edit Boxes .....	225
Using Pop-up Pages .....	226
Understanding Pop-up Pages .....	226
Defining Pop-up Pages .....	229
Associating Pop-up Pages with Fields .....	230
Enabling Message Catalog Pop-up Pages .....	231
Using Subpages .....	236
Understanding Subpages .....	237
Defining Subpages .....	237
Inserting Subpages Into Pages .....	238
Specifying Informational Subpage Labels .....	238
Using Secondary Pages .....	239
Understanding Secondary Pages .....	239
Understanding Modal Secondary Pages .....	241
Defining Secondary Pages .....	242
Inserting Secondary Page Controls .....	244
Associating Secondary Pages With Primary Pages .....	245
Using HTML Areas .....	246
Understanding HTML Area Controls .....	246
Inserting HTML Areas .....	248
Populating HTML Areas .....	249
Changing HTML Area Labels .....	250
Using Push Buttons and Links .....	250
Understanding Push Buttons and Links .....	250
Inserting Push Buttons or Links .....	251

Specifying Type Properties for Push Buttons or Links .....	252
Specifying Label Properties for Push Buttons or Links .....	256
Specifying Use Properties for Push Buttons or Links .....	260
Programmatically Change Hover Text for Push Buttons or Links .....	263

## Chapter 10

<b>Using Scroll Areas, Scroll Bars, and Grids .....</b>	<b>265</b>
Understanding Scroll Areas, Scroll Bars, and Grids .....	265
Using Scroll Areas and Scroll Bars .....	265
Understanding Scroll Areas and Scroll Bars .....	266
Setting Scroll Area General Attributes .....	267
Setting Scroll Area Label Properties .....	269
Setting Scroll Area Use Properties .....	276
Setting Scroll Bar Label Properties .....	277
Setting Scroll Bar Use Properties .....	277
Controlling Scrollable Data .....	279
Converting Scroll Bars to Grids .....	279
Using Grids .....	280
Understanding Grids .....	281
Understanding Grid Properties .....	281
Understanding Grid Personalization .....	282
Inserting and Resizing Grid Controls .....	283
Inserting and Manipulating Grid Columns .....	283
Setting Column Properties .....	285
Creating Tabbed Grids .....	287
Freezing Grid Columns .....	288
Using Multiple Grids on a Page .....	289
Enabling and Disabling Grid Personalization .....	289
Setting Grid General Properties .....	290
Setting Grid Label Properties .....	291
Setting Grid Use Properties .....	301
Configuring Embedded Help for Scroll Areas and Grids .....	307
Entering Embedded Help Messages in the Message Catalog .....	308
Associating Help Messages with Scroll Areas and Grids .....	310

## Chapter 11

<b>Creating Page Definitions .....</b>	<b>313</b>
Understanding Page Design and Development .....	313
Level-Based Controls .....	314
Keys for Accessing Pages .....	318

Multiple Occurrences of Data .....	319
Prompt Fields .....	320
Derived and Work Fields .....	322
Sensitive Data .....	323
Hidden Pages .....	323
Upgrade Considerations .....	323
End-User Page Personalization .....	323
Using Page Development Tools .....	324
Using Page Definition Menus .....	324
Using Page Definition Toolbars .....	325
Using the Page Layout Grid .....	327
Creating New Page Definitions .....	328
Cloning Page Definitions .....	329
Creating a Blank Page .....	329
Adding Page Controls .....	329
Dragging Field Definitions From a Record Onto a Page .....	330
Dragging Record Fields From the Project Workspace Onto a Page .....	330
Dragging Record Definitions Onto a Page .....	330
Using the Page Control Toolbar .....	331
Using the Insert Menu .....	332
Manipulating Page Controls .....	332
Selecting Controls .....	332
Resizing Controls .....	333
Using the Page Control Inspector .....	333
Deleting Controls .....	334
Moving Controls on the Same Page .....	335
Copying and Moving Controls to Another Page .....	335
Positioning Page Control Labels .....	335
Setting Page Field Properties for Controls .....	337
Setting Record Properties .....	338
Setting Label Properties .....	344
Setting Use Properties .....	348
Setting General Properties .....	354
Setting Options Properties .....	356
Modifying the Rich Text Editor User Interface .....	358
Creating Custom Configurations .....	359
Creating Custom Template Files .....	360
Creating Custom Style Files .....	360
Creating Custom Plug-In Files .....	361
Modifying the Default Image Storage Location .....	361
Creating Display Control and Related Fields .....	362
Using Display Control and Related Fields .....	362
Creating Related Edit Fields .....	363
Creating Invisible Control Fields .....	364
Ordering and Spacing Page Controls .....	365

Understanding the Page Definition Order Tab .....	365
Understanding Page Control Order and Spacing .....	366
Ordering Controls Visually .....	366
Ordering Controls Logically .....	367
Testing Page Control Order .....	368
Applying Control Order Rules .....	368
Changing Control Order Using the Order Tab .....	369
Finding a Field on a Page Definition .....	370
Using Default Order .....	370
Spacing Controls on Pages .....	370
Designing Inquiry Pages .....	371
Aligning Page Controls and Page Control Labels .....	371
Maximizing Performance .....	372
Accessing PeopleCode in Page Definitions .....	373
Viewing Pages in the Browser .....	375
Working With the View in Browser Feature .....	375
Changing the Default Browser .....	377
Generating HTML .....	378
Viewing the HTML for a Page .....	379
Identifying Page Definitions Online .....	379
Producing Pages .....	381
Understanding Page Production .....	382
Changing General Page Information .....	382
Changing Page Use Information .....	383
Saving Pages .....	386
Renaming and Deleting Pages .....	386
Printing Page Definitions .....	387
Altering Page Setup .....	387
Reading a Page Definition Report .....	388
Working with Pages in Foreign Languages .....	390
Understanding Working with Pages in a Foreign Language .....	390
Performing a Save As Using Foreign Language Pages .....	390
Deleting Foreign Language Pages .....	391

## Chapter 12

<b>Creating Component Definitions .....</b>	<b>393</b>
Understanding Component Definitions .....	393
What Component Definitions Manage .....	393
The Component Buffer .....	394
Defining Components .....	395
Understanding the Component Definition Window .....	395
Creating New Component Definitions .....	397

Opening Existing Component Definitions .....	397
Adding Pages to Components .....	397
Reordering Pages in a Component .....	398
Copying or Moving Pages to Another Component .....	399
Setting Page Attributes .....	399
Validating Components .....	401
Specifying Market-Specific Components .....	402
Locating Component Definition References In Menus .....	403
Setting Component Properties .....	403
Opening the Component Properties Dialog Box .....	403
Setting General Properties .....	403
Setting Use Properties .....	404
Setting Internet Properties .....	406
Using Search Records .....	413
Understanding Search Records .....	413
Understanding Search Records and Recent (Persistent) Search Results .....	415
Creating Search Records .....	418
Overriding Search Records .....	419
Accessing the Message Catalog .....	419
Enabling the Expert Entry Feature in the Security Component .....	420
Understanding the Expert Entry Feature .....	420
Enabling Expert Entry in Security .....	421

## Chapter 13

<b>Creating Menu Definitions .....</b>	<b>423</b>
Understanding Menus .....	423
Defining Custom Component Menus .....	423
Creating Custom Component Menus .....	424
Setting Menu Item Properties .....	425
Setting General Properties .....	426
(Optional) Setting Use Properties .....	426
Working With Menu Definitions .....	426
Adding Components to a Menu .....	427
Renaming Menu Definitions .....	427
Copying Menu Definitions .....	427
Deleting Menu Items .....	428
Printing Menu Definitions .....	428
Setting Up Menu Security .....	428
Importing Menu Groups Into Portals .....	429
Defining Pop-up Menus .....	429
Understanding Pop-up Menus .....	429
Creating Pop-up Menus .....	430

Defining Transfer Menu Items .....	432
 <b>Chapter 14</b>	
<b>Using the Registration Wizard .....</b>	<b>435</b>
Understanding the Registration Wizard .....	435
Common Elements Used in This Chapter .....	435
Component Registration .....	436
Registering Components and iScripts .....	436
Registering Components in Multiple Portals .....	442
Registering Mobile Pages .....	442
 <b>Chapter 15</b>	
<b>Creating Style Sheet Definitions .....</b>	<b>445</b>
Understanding Style Sheets and Style Classes .....	445
Overview of Style Sheets and Style Classes .....	445
Style Sheet Types .....	446
Style Classes .....	449
Creating Standard and Sub Style Sheets .....	451
Creating Standard Style Sheets or Sub Style Sheets .....	451
Renaming Standard Style Sheets or Sub Style Sheets .....	452
Deleting Standard Style Sheets or Sub Style Sheets .....	452
Inserting Sub Style Sheets into Style Sheets .....	453
Inserting Style Classes into Style Sheets .....	453
Inserting Override Style Classes into Style Sheets .....	454
Setting Style Class Attributes .....	454
Understanding Class Attributes .....	455
Understanding Generic and Installed Fonts .....	455
Accessing Class Attributes .....	456
Specifying Fonts .....	456
Setting Font Attributes for a Specific Language .....	458
Specifying Spacing and Alignment .....	458
Specifying Background Attributes .....	461
Referencing a Background Image URL .....	464
Specifying Border Attributes .....	464
Specifying Margins .....	465
Specifying Miscellaneous Attributes .....	467
Working With Free Form Sub Style Sheets .....	470
Creating Free Form Sub Style Sheets .....	471
Inserting Free Form Sub Style Sheets into Standard Style Sheets .....	471
Converting Style Sheets to Free Form Sub Style Sheets .....	471

Renaming Free Form Sub Style Sheets .....	472
Deleting Free Form Sub Style Sheets .....	472
Including Meta Variables and Comments in Free Form Sub Style Sheets .....	472
Changing Font and Color Settings in Free Form Sub Style Sheets .....	473
Working with the Default Style Sheet .....	475
Configuring PSSTYLEDEF as the Default Style Sheet .....	475
Configuring PSSTYLEDEF_SWAN as the Default Style Sheet .....	477
Configuring Other Default Style Sheets .....	478
Overriding Default Page Style Properties .....	479
Overriding the Default Page Style Sheet .....	479
Overriding the Default Page Background Style .....	480
Overriding the Default Page Field Styles .....	480
Modifying Page Tabs .....	481
Understanding Tabs .....	481
Creating Tab Images .....	483
Customizing Page Tab Style Sheets .....	484
Modifying Push Buttons in SWAN Style Sheets .....	485
Understanding Push Button in the SWAN Style Sheet .....	485
Creating SWAN Style Sheet Push Button Images .....	485
Customizing SWAN Push Buttons in HTML Areas .....	486

## Chapter 16

<b>Creating Image Definitions .....</b>	<b>487</b>
Understanding Image Definitions .....	487
Understanding the Image Catalog .....	488
Creating Image Definitions .....	489
Creating Image Definitions .....	489
Setting Image Properties .....	490
Importing and Creating Alternate Image Types .....	491
Opening Image Definitions .....	491
Updating Image Definitions .....	491
Updating Image Definitions .....	492
Changing an Image Display Size .....	492
Specifying an Image Storage Format .....	492
Converting Images to JPGs .....	493
Consolidating Images .....	494

## Chapter 17

<b>Creating HTML Definitions .....</b>	<b>497</b>
Understanding HTML Definitions .....	497

Creating HTML Definitions .....	497
Opening HTML Definitions .....	498
Referencing HTML Definitions Dynamically .....	498

## Part 3

### Maintaining Data

#### Chapter 18

<b>Constructing File Layouts and Performing Data Interchanges .....</b>	<b>501</b>
Understanding File Layouts .....	501
File Layouts .....	501
Field Formats in a Flat File .....	502
FIXED Format Files .....	503
CSV Format Files .....	504
XML Format Files .....	506
Constructing File Layouts .....	507
Creating New File Layout Definitions .....	507
Adding File Records and File Fields .....	508
Reordering File Layout Nodes .....	510
Naming File Layouts, Records, and Fields .....	510
Applying Field Formats .....	511
Specifying File Layout, Record, and Field Properties .....	512
Specifying File Layout Properties .....	513
Specifying File Record Properties .....	515
Specifying File Field Properties .....	517
Performing Data Interchanges .....	521
Understanding the Import Process .....	521
Previewing Input Data .....	522
Generating and Running the Import Program .....	525
Exporting Data .....	525
Producing Sample File Layouts .....	525
Creating File Layout Definitions .....	526
Adjusting File Layout Properties .....	527
Inserting Segments and Fields .....	528

#### Chapter 19

<b>Using Command Line Parameters .....</b>	<b>531</b>
--	------------



Understanding PeopleSoft Application Designer Command Line Parameters .....	531
Starting the PeopleSoft Application Designer Command Line Interface .....	532
Building Projects .....	534
Copying Definitions .....	536
Comparing Definitions for Upgrade Using the Command Line .....	542

## Chapter 20

<b>Using Change Control .....</b>	<b>551</b>
Understanding Change Control Features .....	551
Change Control Locking .....	551
Change Control History .....	553
Change Control Stamping .....	554
Understanding Implementation Considerations .....	555
Understanding Change Control Information Reporting .....	556
PeopleTools Definition Types .....	557
PeopleCode Definition Types .....	560
Change Control Supported Definition Types .....	561
Setting Up Change Control .....	562
Understanding Change Control Security .....	562
Appointing a Change Control Administrator .....	563
Enabling or Disabling Change Control .....	563
Using Projects .....	564
Understanding Projects .....	564
Using Multiple Databases for Development .....	565
Using Distributed Development Environments .....	565
Using Change Control .....	566
Locking and Unlocking Definitions .....	566
Viewing Locked Definitions .....	568
Inserting Comments .....	568
Deleting Page Definitions .....	570
Viewing Change Control History .....	570

## Appendix A

<b>Understanding Functional Indexes .....</b>	<b>573</b>
Functional Indexes for Microsoft SQL Server 2000 .....	573
Functional Indexes for DB2/UDB for Linux, UNIX, and Windows .....	575
Functional Indexes for Informix .....	575

**Appendix B**

<b>Referencing Selections for Command Line Parameters .....</b>	<b>579</b>
Database Type Selections .....	579
Definition Type Selections .....	580
Language Selections .....	586
Available Release Numbers .....	588

**Appendix C**

<b>Using Default Style Classes .....</b>	<b>591</b>
Default Style Classes .....	591

**Appendix D**

<b>Modifying Classic Style Sheet Tabs .....</b>	<b>611</b>
Understanding Classic Tab Definitions .....	611
Modifying Classic Style Sheet Tabs .....	612
Creating Tab Images .....	612

**Appendix E**

<b>Using Images in PeopleSoft Applications .....</b>	<b>617</b>
Understanding Image-Related Elements .....	617
Using PeopleTools Image Definitions .....	618
Generating a List of Images in the Image Catalog .....	622
Viewing All Images in the Image Catalog .....	623

**Appendix F**

<b>Performing Bulk Operations .....</b>	<b>625</b>
Understanding Bulk Operations .....	625
Performing Bulk Insertions .....	625
Common Elements Used in This Section .....	626
Inserting Fields into Records .....	626
Inserting Record Fields onto Pages .....	626
Inserting Fields into Records and onto Pages .....	627

Performing Bulk Modifications .....	627
Modifying Record Fields .....	628
Modifying Page Fields .....	628
Reordering or Resizing Grid Fields .....	628
Changing Page Field Attributes .....	629
Performing Bulk Deletions .....	629

## Appendix G

<b>Running PeopleTools Cross-Reference Reports .....</b>	<b>631</b>
Understanding Cross-Reference Reports .....	631
Running Crystal Reports .....	633

## Appendix H

<b>Working With Records Used in the Mobile Synchronization Framework .....</b>	<b>635</b>
Understanding Mobile Synchronization Framework Records .....	635
Configuring Records for the Mobile Synchronization Framework .....	635
Setting Record Properties .....	636
Working with Required Fields and Default Values .....	636
Example: Building Records for the Mobile Synchronization Framework .....	637

## Appendix I

<b>Creating Custom Plug-In Files for the Rich Text Editor .....</b>	<b>639</b>
Creating Custom Buttons .....	639
Using Images with Plug-Ins .....	640
Adding Multilingual Support .....	640
Adding Plug-Ins to the Configuration File .....	641
Specifying the Plug-In File ID Value .....	642

## Appendix J

<b>Configuring Rich Text-Enabled Fields to Store Accented Western European Characters .....</b>	<b>643</b>
Enabling Field-Level Support For Accented Characters .....	643
Enabling System-Wide Support For Accented Characters .....	644
Sample config_psft.js File .....	645

**Appendix K**

**Creating Custom SmartNavigation Pages ..... 647**

Understanding the Elements of SmartNavigation Pages ..... 647

**Appendix L**

**Configuring Embedded Help to Display UPK Content ..... 651**

Modifying the UPK Player Package Link ..... 651

Adding UPK-Based URLs to the Application ..... 652

Embedding UPK-Based URLs in Messages ..... 653

Adding Embedded Help to Page Controls ..... 653

**Index ..... 655**

# Oracle's PeopleSoft Application Designer

## Preface

This chapter discusses PeopleSoft Application Designer.

---

## Oracle's PeopleSoft Application Designer

PeopleSoft Application Designer enables application developers to create PeopleSoft pages that can be opened from a browser and are available over the internet. This PeopleBook describes the fundamental elements of developing applications for the PeopleSoft Pure Internet Architecture by using PeopleSoft Application Designer. It discusses defining fields, records, and pages; adding specific page controls to PeopleSoft pages; combining pages into components; applying style sheets; creating menus; building SQL tables and views, and upgrading application versions.

For information that is specific to your application, please refer to the application PeopleBook for your specific PeopleSoft product.

---

## PeopleBooks and the PeopleSoft Online Library

A companion PeopleBook called *PeopleBooks and the PeopleSoft Online Library* contains general information, including:

- Understanding the PeopleSoft online library and related documentation.
- How to send PeopleSoft documentation comments and suggestions to Oracle.
- How to access hosted PeopleBooks, downloadable HTML PeopleBooks, and downloadable PDF PeopleBooks as well as documentation updates.
- Understanding PeopleBook structure.
- Typographical conventions and visual cues used in PeopleBooks.
- ISO country codes and currency codes.
- PeopleBooks that are common across multiple applications.
- Common elements used in PeopleBooks.
- Navigating the PeopleBooks interface and searching the PeopleSoft online library.
- Displaying and printing screen shots and graphics in PeopleBooks.
- How to manage the locally installed PeopleSoft online library, including web site folders.
- Understanding documentation integration and how to integrate customized documentation into the library.
- Application abbreviations found in application fields.

You can find *PeopleBooks and the PeopleSoft Online Library* in the online PeopleBooks Library for your PeopleTools release.

## **Part 1**

# **Getting Started with PeopleSoft Application Designer**

## **Chapter 1**

### **Getting Started with PeopleSoft Application Designer**





## Chapter 1

# Getting Started with PeopleSoft Application Designer

This chapter provides an overview of PeopleSoft Application Designer and discusses the implementation tasks necessary to work with this tool.

---

## PeopleSoft Application Designer Overview

PeopleSoft Application Designer is the core development tool used to build and modify PeopleSoft applications. PeopleSoft Application Designer enables you to build a variety of definitions, including:

- Fields
- Records (Tables)
- Pages
- Components
- Menus
- PeopleCode
- SQL

---

## PeopleSoft Application Designer Implementation

This section discusses:

- Installation-level steps.
- Main application development steps.

### Installation-Level Steps

This table lists the steps you must complete before you can begin using PeopleSoft Application Designer:

<b>Step</b>	<b>Explanation</b>	<b>Reference</b>
Set up security.	For developing and customizing PeopleSoft applications, you need access to the definitions involved, such as fields, records, pages, and so on. Your administrator can provide access by adjusting your definition security settings. Your administrator also might need to grant general PeopleTools permissions for maintaining and upgrading data. These permissions include Build and Data Administration, Change Control, and Upgrade.	See <i>PeopleTools 8.52: Security Administration</i> , "Implementing Definition Security."  See <i>PeopleTools 8.52: Security Administration</i> , "Setting Up Permission Lists," Setting PeopleTools Permissions.
Set up workstations.	PeopleSoft Application Designer is a Microsoft Windows-based program. To run PeopleSoft Application Designer, you must have a supported version of Windows with ample CPU and memory resources. Also, you must use PeopleSoft Configuration Manager to configure your workstation with connectivity to the appropriate database and to set all required environment variables.	See <i>PeopleTools 8.52: System and Server Administration</i> , "Using PeopleSoft Configuration Manager," Configuring Developer Workstations.

## Main Application Development Steps

This table presents the main steps for creating a PeopleSoft application:

<b>Step</b>	<b>Reference</b>
1. Design application.	Review your in-house development cycle and documentation.
2. Create fields.	See <a href="#">Chapter 4, "Creating Field Definitions," page 63.</a>
3. Create record definitions.	See <a href="#">Chapter 5, "Planning Records, Control Tables, and TableSets," page 97.</a> See <a href="#">Chapter 6, "Creating Record Definitions," page 115.</a>
4. Build SQL tables.	See <a href="#">Chapter 7, "Administering Data," page 155.</a>
5. Create pages.	See <a href="#">Chapter 8, "Designing Pages for Optimal Performance," page 199.</a> See <a href="#">Chapter 9, "Using Page Controls," page 207.</a> See <a href="#">Chapter 11, "Creating Page Definitions," page 313.</a>
6. Create components.	See <a href="#">Chapter 12, "Creating Component Definitions," page 393.</a>

<b>Step</b>	<b>Reference</b>
7. Register components.	See <a href="#">Chapter 14, "Using the Registration Wizard," page 435.</a>
8. Test the application.	Review your in-house testing procedures and documentation.

### ***Other Sources of Information***

This section provides information you should consider before you begin to use PeopleSoft Application Designer. In the planning phase of your implementation, take advantage of all Oracle sources of information, including the installation guides and troubleshooting information. A complete list of these resources appears in the preface in About These PeopleBooks, with information about where to find the most current version of each.

### ***See Also***

*PeopleTools 8.52: Getting Started with PeopleTools*, "Getting Started with Enterprise PeopleTools Preface"



## **Part 2**

# **Developing and Customizing PeopleSoft Applications**

### **Chapter 2**

#### **Using PeopleSoft Application Designer**

### **Chapter 3**

#### **Working With Projects**

### **Chapter 4**

#### **Creating Field Definitions**

### **Chapter 5**

#### **Planning Records, Control Tables, and TableSets**

### **Chapter 6**

#### **Creating Record Definitions**

### **Chapter 7**

#### **Administering Data**

### **Chapter 8**

#### **Designing Pages for Optimal Performance**

### **Chapter 9**

#### **Using Page Controls**

### **Chapter 10**

#### **Using Scroll Areas, Scroll Bars, and Grids**

### **Chapter 11**

#### **Creating Page Definitions**

### **Chapter 12**

#### **Creating Component Definitions**

**Chapter 13**  
**Creating Menu Definitions**

**Chapter 14**  
**Using the Registration Wizard**

**Chapter 15**  
**Creating Style Sheet Definitions**

**Chapter 16**  
**Creating Image Definitions**

**Chapter 17**  
**Creating HTML Definitions**

## Chapter 2

# Using PeopleSoft Application Designer

This chapter provides an overview of PeopleSoft Application Designer and discusses how to:

- Use PeopleSoft Application Designer.
- Work with definitions.
- Use the programming language editors.
- Use miscellaneous definitions.
- Use the PeopleSoft Application Designer spell check feature.
- Build and maintain data.

---

## Understanding PeopleSoft Application Designer

Every PeopleSoft Pure Internet Architecture application contains a collection of related definitions that work together for a specific purpose. Developing and adapting PeopleSoft applications is a step-by-step process in which you define and build the definitions, establish relationships among the definitions, implement security, run your PeopleSoft application in an internet browser, and test every aspect thoroughly. You use one interactive tool, PeopleSoft Application Designer, for the majority of these activities. PeopleSoft Application Designer is an integrated development environment that enables you to work with the numerous definitions of a business application in a single work area.

---

**Note.** Your user interface might appear differently than what you see in this PeopleBook. The look and feel of PeopleSoft Application Designer varies based on the release of your Microsoft ® Windows operating system.

---

---

## Using PeopleSoft Application Designer

This section provides an overview of the eight steps of application development and discusses how to:

- Start PeopleSoft Application Designer.
- Use the PeopleSoft Application Designer window components.
- Set PeopleTools Options

## Understanding the Eight Steps of Application Development

Application development is a systematic process. Between the design and implementation of the application, you define and build application definitions, establish relationships, implement security, and test every aspect thoroughly. When developing in the PeopleSoft environment, it is helpful to look at the development phase as an eight step process:

1. Design the application.
2. Create field definitions.
3. Create record definitions.
4. Build application data tables.
5. Create page definitions.
6. Define components.
7. Register components.
8. Test the application.

### ***Step One - Designing the Application***

Create a development plan. The first phase of development in any environment is always planning. Designers might spend 80 percent of their time designing a sound plan before they work on the first step. In PeopleSoft applications, you will need to fully understand the other seven steps of development to design a good plan. When you finish the development plan, you should have a business requirements document that specifies the tables required and the relationships among them, a list of required definitions (fields, records, and so on) complete with detailed specifications and naming conventions, mock-ups of page layouts, and a navigational hierarchy that will give users access to the pages you create.

### ***Step Two - Creating Field Definitions***

Field definitions in PeopleSoft are slightly different from other environments. Designers define fields apart from any tables they may belong to and ensure that they can be reused in any number of separate database tables. The field definition is the basis for all the steps that follow.

### ***Step Three - Creating Record Definitions***

PeopleTools-based applications are table-based systems. When creating a record definition, you are actually defining two layers of information: the record level and the field level. At the record level, you define the structure of the database table, and the ultimate purpose of the record definition, and how the system will use the definition. At the field level, you define the characteristics of the fields and any online processing that you want to occur behind the scenes when a user enters data.



### ***Step Four - Building Application Data Tables***

After you define a record definition, you must create or build the underlying SQL tables that house the application data that users will enter online in the production environment. This build process is the centerpiece of the data administration features in PeopleSoft Application Designer. The build process takes all of the properties of a record definition and creates a database table from them. Unless you complete this step, no table will exist in the database to accept the application data.

### ***Step Five - Creating Page Definitions***

Pages are the user interface for the application. Pages present data to the users and enable them to enter data into the application data table created in the previous step. In this step, you add page controls to a page, associate the page controls with record fields, and configure appearance of the page. The system displays the page to users in a browser window as HTML.

### ***Step Six - Defining Components***

Component definitions represent complete business transactions in PeopleSoft applications and are central to navigation and security access. A component holds one or more pages that the user navigates between using tabs or links. In this step, you define the search record and search attributes, the processing mode, and the toolbar buttons that appear at the bottom of each page in the component.

### ***Step Seven - Registering Components***

The Registration Wizard makes registering a component easy. In this step, you attach the component to a menu, assign security access for users, and place navigational links in the registry. After you register a component, the user interface is complete and you can view the application in the browser.

### ***Step Eight - Testing the Application***

The final step of application development is to thoroughly test the application before delivering the finished product to users. You must test all aspects of the application by opening a transaction in your browser and entering test data into the application. As issues arise, you can cycle back through each of the previous steps to resolve the issues.

## **Starting PeopleSoft Application Designer**

Open PeopleSoft Application Designer from the Start menu. Alternatively, if you have set up a shortcut on your desktop, you can open the application from there. At the PeopleSoft Signon dialog box, enter the Database Name or Application Server Name (depending on your connection type), User ID, and Password. PeopleTools opens automatically to PeopleSoft Application Designer.

---

**Note.** Specific PeopleSoft utilities require you to set an environment variable, PS\_SERVER\_CFG, to run properly. However, PeopleSoft Application Designer is not compatible if PS\_SERVER\_CFG is set. Before you start PeopleSoft Application Designer, you must ensure that PS\_SERVER\_CFG is not set. A convenient way to do this is to use a DOS batch file to unset PS\_SERVER\_CFG, launch PeopleSoft Application Designer, and then—after PeopleSoft Application Designer exits—reset PS\_SERVER\_CFG to its previous value.

---

In addition to launching PeopleSoft Application Designer from the Start menu, you can also do so using the command line syntax and the available parameters listed there.

PeopleSoft Application Designer offers a variety of command line parameters that you can use to control the database to which it connects. By using these parameters, you can automatically navigate to the part of the system that you need.

Use the values from the following tables to specify specific parameters.

### **Syntax**

The following command line syntax applies to all actions identified in this chapter:

PSIDE [-parameter value [-parameter value . . .]]

Use this command line syntax to start PeopleSoft Data Mover:

PSDMT [-parameter value [-parameter value . . .]]

You can include as many or as few parameters as you need.

Each parameter starts with a hyphen (-) or a forward slash (/). The value for each parameter follows the hyphen or slash, separated by zero or more spaces. In general, the value does not need to have quotation marks around it, even if it has internal spaces, because the system treats all text following the parameter as part of the value up to the next parameter or the end of the command line.

---

**Note.** You must enclose a value in quotation marks only when it includes a hyphen or forward slash or to include leading or trailing spaces. If the value itself includes a quotation mark character, then precede the double quotation mark with a backslash (\).

---

### **Available Parameters**

The available parameters for PSIDE.EXE fall into two general categories:

- Parameters that provide login information.
- Parameters that set general options.

If the command line includes login parameters, it uses them only if no PeopleSoft applications are currently running. If a PeopleSoft application is already running, then the system starts a new instance by using the same login information as the active instance.

This table lists all of the available parameters:

<i><b>Parameter</b></i>	<i><b>Value</b></i>	<i><b>Description</b></i>
-CT	Database type	Enter the type of database to which you are connecting. Valid values are ORACLE, INFORMIX, SYBASE, MICROSOFT, DB2ODBC, and DB2 UDB for Linux, UNIX, and Microsoft (note the spelling of MICROSOFT).

<i><b>Parameter</b></i>	<i><b>Value</b></i>	<i><b>Description</b></i>
-CS	Server name	Enter the name of the database server for the database to which you are connecting. This setting is required for some database types.
-CD	Database name	Enter the name of the database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.
-CO	User ID	Enter the PeopleSoft user ID used for sign in.
-CP	Password	Enter the password for the specified user ID.
-CI	Connect ID	Enter the ID that is used to connect to the database server.
-CW	Password	Enter the password for the specified connect ID.
-SS	NO	Use to suppress the appearance of the PeopleSoft splash screen. Any other value will cause the splash screen to appear.
-SN	NO	Use to suppress the sound that plays when you sign in to the PeopleSoft system. Any other value will allow the sound to play.
-QUIET	(none)	Use to run in quiet mode so that no message boxes appear.
-HIDE	(none)	Use to hide the PeopleSoft Application Designer interface.

### **Example**

This command line statement starts PeopleSoft Application Designer and signs the user in to the DEP7TST database:

```
PSIDE -CT MICROSOFT -CS SEPNTDB05 -CD EP7TST -CO VP1 -CP VP1
```

## Using the PeopleSoft Application Designer User Interface

This table describes the six main elements that comprise the PeopleSoft Application Designer user interface:

<i>Element</i>	<i>Description</i>
Title bar	Displays the name of the open project and active definition.
Menu	Provides access to PeopleSoft Application Designer items and features.
Toolbar	Displays buttons that are useful in editing the active definition.
Project workspace	Provides a graphical representation of the components that are contained in a project. The project workspace has two folder tabs: <ul style="list-style-type: none"> <li>• Development View.</li> <li>• Upgrade View.</li> </ul>
Definition workspace	Displays individual definitions that you open.
Output window	Contains the output text from PeopleSoft Application Designer operations, such as Build (SQL Create and Alter), Find Definition References, Upgrade, Results, Validate, and PeopleCode Log.

---

**Note.** You should maximize the PeopleSoft Application Designer window to make optimal use of the different workspaces.

---

### ***Application Designer Menus***

This topic reviews the general PeopleSoft Application Designer menus. Specific menus appear for each definition when it is open and active. Those menus are discussed in the topic about each definition.

---

**Note.** The menu items that appear on each menu often depend on the type of definition that is open in the definition workspace. Those menu items are described in the corresponding chapters.

---

In addition to the standard Microsoft Windows File menu items, several PeopleTools-specific items appear in the File menu. This table lists the PeopleTools-related File menu items:

<b><i>File Menu Items</i></b>	<b><i>Usage</i></b>
Save Project	Save the project.
Save Project As	Save the active project definition. This menu item enables you to change the name of the project when you save it.
Print Project	Print an Upgrade Project report for all the elements in the open project.
Merge Projects	Create a new project from selected definitions of two existing projects.
Project Properties	Edit properties for the open project.
Definition Properties	Provide general comments and change attributes for the active definition. You can change attributes for Use, Type, and Internet properties, depending on the definition type.
View Report	View a report of the definition in an open window in the workspace.
Report from File, Print Preview	View a print preview of a selected report file that is saved to a disk.
Report from File, Print	Print a selected report file from a disk.
Report from File, View Report	View a report from a selected report file that is saved to a disk.
List of last opened definitions	View and open one of the last few definitions you opened. You can configure the maximum number of definitions that appear in the list.  See <a href="#">Chapter 2, "Using PeopleSoft Application Designer," Setting PeopleTools Options, page 22.</a>

In addition to the standard Microsoft Windows-based menu items, such as Cut and Paste, two PeopleTools-specific menu items appear in the Edit menu. This table describes their use:

<b>Edit Menu Items</b>	<b>Usage</b>
Find Definition References	Search for all related definitions and references (fields, records, pages, PeopleCode, application packages, and so on) to the current open definition. The results appear on the Find Definition References tab of the Display window.
Find In	Search through all PeopleCode, SQL programs, HTML definitions, and Free form sub style sheets for a text string that you describe in a dialog box. You can also specify which type of PeopleCode and SQL programs to search.

This table describes the use of View menu items:

<b>View Menu Items</b>	<b>Usage</b>
View Definition	View a definition that is referenced in the active definition. For example, if you highlight a record field in a record definition, then selecting this option opens the selected field definition.
View PeopleCode	View PeopleCode characteristics of the active definition.
Filtering	Select a filter option for your view of upgrade: <ul style="list-style-type: none"> <li>• No Filtering.</li> <li>• Selected for Upgrade Action.</li> <li>• Not Selected for Upgrade Action.</li> <li>• Custom Filtering.</li> </ul>
Toolbar	Activate and deactivate the toolbar.
Status Bar	Activate and deactivate the status bar.
Project Workspace	Activate and deactivate the project workspace display.
Output Window	Activate and deactivate the output window display.

This table describes the use of Insert menu items:

<b><i>Insert Menu Items</i></b>	<b><i>Usage</i></b>
Current Definition into Project	Insert the active definition into the project.
Definitions into Project	Display a dialog box to insert definitions into the project.

This table describes the use of Build menu items:

<b><i>Build Menu Items</i></b>	<b><i>Usage</i></b>
Current Definition	Display a dialog box to build or alter the record definition that is currently active in the definition workspace.
Project	Display a dialog box to build (SQL Create and SQL Alter) all project records.
Settings	Display a dialog box in which you set parameters that apply to build operations.

This table describes the use of Debug menu items:

<b><i>Debug Menu Items</i></b>	<b><i>Usage</i></b>
PeopleCode Debugger Mode	Start the PeopleCode Debugger, which invokes a debugging session and causes a full Debug menu to appear. When in Debugger mode, a Local Variables pop-up window appears in the definition workspace.  See <i>PeopleTools 8.52: PeopleCode Developer's Guide</i> , "Debugging Your Application."

This table describes the use of Tools menu items:

<b><i>Tools Menu Items</i></b>	<b><i>Usage</i></b>
Validate Project	Validate attributes of the active project.

<b><i>Tools Menu Items</i></b>	<b><i>Usage</i></b>
Validate Database for WSRP	Validates all of the appropriate definitions within a database for WSRP compliance.  <i>See PeopleTools 8.52: PeopleTools Portal Technologies , "Using WSRP to Consume and Produce Remote Portlets," Validating WSRP Compliance.</i>
Compile All PeopleCode	Validate all of the PeopleCode in the current database.
Compile Project PeopleCode	Validate the PeopleCode in the current project.
Diff / Merge Project PeopleCode	Compare and merge two PeopleCode definitions during an upgrade.
Diff / Merge External Text	Compare and merge two versions of an external text file (such as a Data Mover script or a SQL script) during an upgrade.
Compare and Report	Compare two projects and report on the differences.
Copy Project	Copy the current project to a database or a file, or export a project from a file to your database.
Data Administration	Open dialog boxes that describe space allocation in the database.
Change Control	Open dialog boxes to view history or set systemwide change control options as the administrator.
Upgrade	Upgrade an application project.
Translate	Translate base and operator language databases.
Miscellaneous Definitions	Set custom field formats, toolbars, colors, styles, access profiles, and file type codes.  <b>Important!</b> Styles are a deprecated feature. You must maintain styles by using style sheet definitions.  <i>See Chapter 15, "Creating Style Sheet Definitions," Understanding Style Sheets and Style Classes, page 445.</i>



<b><i>Tools Menu Items</i></b>	<b><i>Usage</i></b>
Bulk Operations	Insert, modify, or delete certain definition types across your entire database in one operation.
Options	Select commands specifying insertion commands, definition language preference, validating options, PeopleCode fonts, definition owner ID, and more.

This table describes the use of Window menu items:

<b><i>Window Menu Items</i></b>	<b><i>Usage</i></b>
Cascade	Cascade all of the definitions that are not minimized in the definition workspace into overlapping layers of definitions with the definition names visible.
Tile	Reorganize and resize all of the definitions that are not minimized in the definition workspace so that they are all visible and do not overlap.
Arrange Icons	Group all of the minimized definitions in the definition workspace.
Close All	Close all of the definitions in the definition workspace.

This table describes the use of Help menu items:

<b><i>Help Menu Items</i></b>	<b><i>Usage</i></b>
Current Window	Provide context-sensitive help that is related to the active window by opening the appropriate PeopleBook in the appropriate location.
PeopleBooks Library	Go to the PeopleSoft Online Library (the online documentation for PeopleTools and your licensed PeopleSoft applications).
About PeopleTools	Display the PeopleTools release level, application release, user ID, database name, database type, and application server name.

## PeopleSoft Application Designer Toolbars

In addition to the standard Microsoft Windows toolbar button items, such as Cut, Paste, and Save, several items are PeopleTools-specific. The command name appears in parentheses following its description.



Open the Definition Properties dialog box (File, Definition Properties).



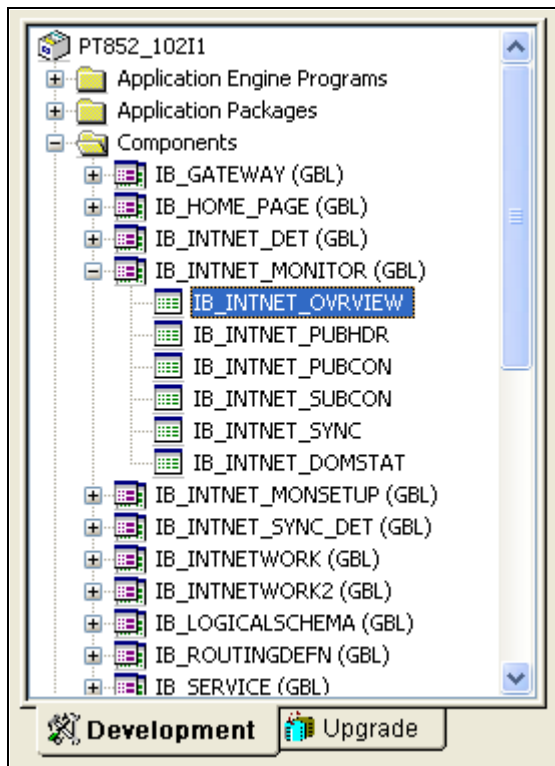
Open the Build dialog box (Build, Current Definition).



Toggle the display of the project workspace (View, Project Workspace).

## Project Workspace

View projects and their associated definitions in the project workspace. A project organizes a collection of definitions in logical groups for easier development, adaptation, maintenance, and upgrade.



Project workspace shown with a project open

Work with a development project or an upgrade project by selecting either the Development tab or the Upgrade tab at the bottom of the project workspace.

The text on the Development and Upgrade tabs might not appear if your monitor is not large enough to display it.

## Definition Workspace

Use the definition workspace to create and modify definitions. You can maximize, minimize, cascade, or tile them in the workspace.

## Output Window

The output window displays status messages of various PeopleSoft Application Designer operations. The output is organized into different categories by using tabs at the bottom of the window.

The text that appears in the output window is context-sensitive, which enables you to select it and perform related operations. For example, if you have a field definition open, select Edit, Find Definition References to list the definitions that reference the active field definition. The list appears in the output window. Double-click any of the related definitions listed to open them.

The output window has several tabs that relate to numerous functions that you can perform with PeopleSoft Application Designer. This table describes the use of the tabs:

<b>Tab</b>	<b>Usage</b>
Build	Create tables, indexes, views, triggers, and alter tables. If you select the <i>Log to Output Window</i> option in the Build settings, then errors, warnings, and informational messages appear here. You can control the level of detail that appears.
Find Definition References	View definitions that other references use or reference. Double-click a definition to open it.
Upgrade	View upgrade actions that are initiated from PeopleSoft Application Designer.
Results	View messages related to project operations, like opening a project or inserting definitions into a project.
Validate	Validate projects and components, and view the results.
Find In	View the Find In dialog box output. Double-click a definition to open it. You can limit the Find In search to a specific project rather than the entire database.
PeopleCode Log	View errors that you receive when you are in the debugger stepping through code.

## Setting PeopleTools Options

You can set preferences for various features in the Options dialog box. Select **Tools, Options** to access the Options dialog box. Seven tabs appear on the dialog box and each tab features configurable options for different areas of PeopleSoft Application Designer. Most of these options are described in further detail in other, more relevant areas of the PeopleBooks documentation. The following topics provides a general description of each tab and a link to further details about the options:

### **Project**

Set various project options.

See [Chapter 3, "Working With Projects," Setting Project Options, page 59.](#)

### **Validate**

Set project validation options, enable the PeopleSoft Application Designer spell check feature, and enable WSRP validation.

See [Chapter 3, "Working With Projects," Validating Projects, page 60.](#)

See [Chapter 2, "Using PeopleSoft Application Designer," Enabling the Spell Check Feature, page 48.](#)

### **Editors**

Set Word Wrap options for the various editors in PeopleSoft Application Designer.

See *PeopleTools 8.52: PeopleCode Developer's Guide*, "Using the PeopleCode Editor."

See *PeopleTools 8.52: PeopleCode Developer's Guide*, "Using the SQL Editor."

### **General**

This table describes the options that you set on the General tab:

<b>Undo/Redo Buffer Depth</b>	Enter the number of most recently performed actions for PeopleSoft Application Designer to store in memory. The default value is <i>10</i> , which mean that you can undo up to 10 immediately previous actions, such as adding or deleting page fields from a page.
<b>Maximum Most Recently Used Definition</b>	Enter the number of recently opened definitions to appear in a list at the bottom of the File menu in PeopleSoft Application Designer. The default value is <i>5</i> .
<b>Report Output Directory</b>	Enter the output destination of the reports that you run in PeopleSoft Application Designer. Reports include upgrade, build, and definition reference reports.

**Enable Translatability Options**

Check to enable the following translatability options:

- When you save a page, PeopleSoft Application Designer warns you if a page control label violates the recommended translation buffer.
- When you add or modify translate values, a warning appears if the Short Name field value exceeds the recommended length of six characters.
- When you add or modify field labels, a warning appears if the Short Name field value exceeds the recommended length of ten characters.
- When you work with page definitions, PeopleSoft Application Designer disables the Static Text tool so that you cannot insert new static labels on pages.

---

**Note.** Translation warnings only appear when your base language value in Configuration Manager is *English*.

---

**Show Hidden Borders**

Check to enable the borders of hidden group boxes and frames to appear as dashed lines indicating the boundary of all hidden group box and frame page control types.

**Owner Id**

Set the default owner ID for new definitions you create. Alternatively, leave the option blank, which is the default, to allow users to select the owner ID they want. The Owner ID helps identify which applications team last made a change to a definition.

**Image**

Set the preferred storage format for newly created image definitions.

See [Chapter 16, "Creating Image Definitions," Specifying an Image Storage Format, page 492.](#)

**Browser**

Enter the base uniform resource indicator (URI) to be used by the browser to locate the application and the node name.

See [Chapter 2, "Using PeopleSoft Application Designer," Configuring Browser Options, page 48.](#)

**See Also**

[Chapter 11, "Creating Page Definitions," Preventing Translation Buffer Overlap, page 336](#)

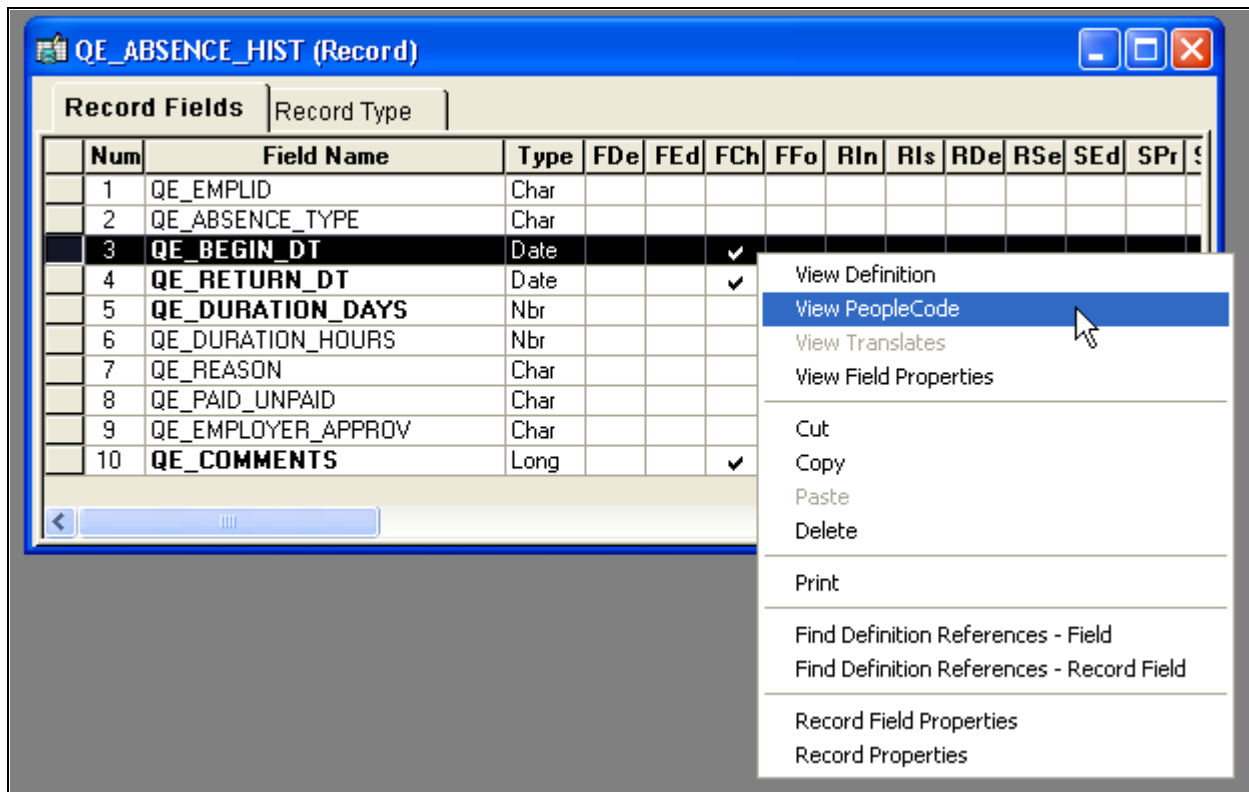
## Using the PeopleCode and Other Programming Language Editors

This section provides an overview of PeopleCode and discusses how to use the various programming language editors.

### Understanding PeopleCode and Other Programming Languages

PeopleCode is the structured programming language built into PeopleTools that extends the functionality of the PeopleTools environment. All PeopleCode programs are associated with a parent definition. PeopleCode programs are considered part of the definitions of their parent components, and you edit them as part of the definitions.

You can associate PeopleCode programs with many items, such as record fields, pages, components, and application messages. You can access any PeopleCode program from its related definition. For example, one of the ways you can access PeopleCode from one of the many places that PeopleCode is stored is to access Record Field PeopleCode from an open record definition.



Example of accessing FieldChange PeopleCode from the QE\_BEGIN\_DT field

To view Record Field PeopleCode from PeopleSoft Application Designer, complete these steps:

1. Open a record definition in PeopleSoft Application Designer.
2. Click the View PeopleCode toolbar button to view the PeopleCode display for the record.
3. Select a field containing PeopleCode in the record definition.

4. Double-click the cell under the column heading for the type of PeopleCode that you want to view.

You can also right-click a definition and use the pop-up menu to access or view the PeopleCode.

---

**Note.** PeopleCode exists on a variety of definitions, such as records, pages, and components. You access PeopleCode through the definition on which it is written.

---

### **See Also**

*PeopleTools 8.52: PeopleCode Developer's Guide, "Accessing PeopleCode and Events"*

## **Using the Programming Editors**

This section discusses these programming editors:

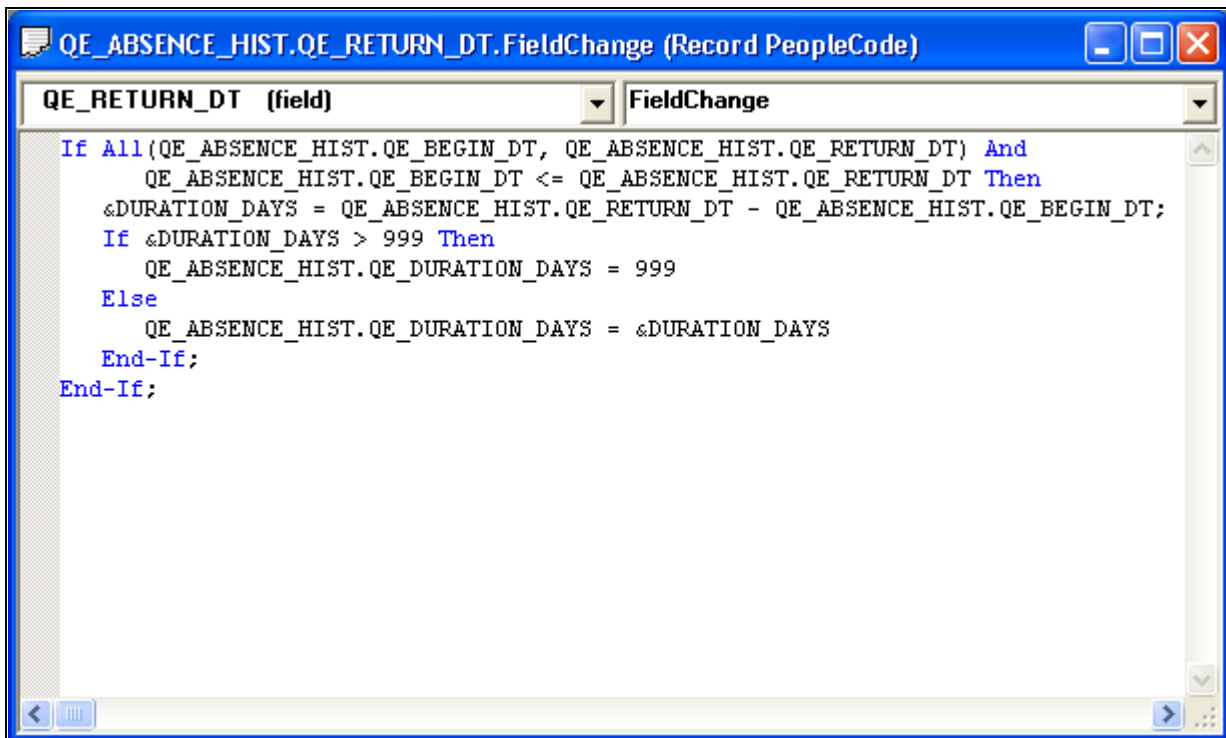
- PeopleCode editor
- SQL editor
- Application packages editor
- HTML editor
- Free form style sheet editor

### ***PeopleCode Editor***

The PeopleCode editor is a text editor that enables you to construct, modify, and review PeopleCode programs and associate them to PeopleSoft definitions and events. The PeopleCode editor enables you to edit and navigate all PeopleCode programs that belong to the same parent definition.

The user interface of the PeopleCode editor consists of two drop-down lists boxes and an area where you enter text. The drop-down list box that is on the left side of the editor displays the current field and other available fields, records, and components. Any field with PeopleCode attached appears in bold. The drop-down list box that is on the right side of the editor displays the current PeopleCode event and other events that are appropriate for that definition. Events already containing PeopleCode for that definition appear in bold. Beneath the drop-down list boxes is where you enter your text. In this area, you can add, delete, and change text and you can use the find and replace function. The PeopleCode editor validates the syntax of your code and the editor automatically formats the code according to PeopleSoft standards when you save the definition.

This example shows the PeopleCode editor:



Example of working in the PeopleCode editor

---

**Note.** You can embed strings in the PeopleCode editor that generate HTML, JavaScript, CSS, and XML markup. However, it is not considered a best practice.

---

### **SQL Editor**

The SQL editor is a text editor that enables you to construct, modify, and review a SQL statement and save it as a reusable SQL definition. Using the SQL editor, you can add, delete, and change text and you can use the find and replace function. The SQL editor formats the text, but the editor does not validate the SQL syntax.

You can access the SQL editor from these definitions:

- Record definitions that are based on SQL and dynamic views.
- PeopleSoft Application Engine actions.
- PeopleCode editor.

### **Application Packages Editor**

From an Application Package you can access the PeopleCode programs that are associated with the classes of the package. The Application Packages editor and the PeopleCode editor interfaces are similar. You can add, delete, and change text; you can use the find and replace function; and you can validate the syntax. When you save your application package, the code is automatically formatted, just as it is in the PeopleCode editor.



### **HTML Editor**

The HTML editor interface offers some of the same functionality as the other editors. In the HTML editor you can add, delete, and change text; and you can use the find and replace function. While the editor does format certain elements of your code, such as variables, methods, and strings, the editor does *not* validate the syntax. Because the HTML editor is free form, you can enter any type of text in it because there is no validation of any kind. However because the HTML editor output is used in web pages, typically, you enter only the following markup languages: HTML, JavaScript, CSS, and XML

### **Free Form Style Sheet Editor**

The free form style sheet editor interface enables you to create text-based sub style sheets that enable you to take advantage of Cascading Style Sheets - Level 2 (CSS2), AJAX, and DHTML. In the free form style sheet editor you can add, delete, and change text; and you can use the find and replace function. The editor does *not* format the text or validate the syntax.

### **See Also**

*PeopleTools 8.52: PeopleCode Developer's Guide, "Using the SQL Editor"*

*PeopleTools 8.52: PeopleCode Developer's Guide, "Using the PeopleCode Editor"*

*PeopleTools 8.52: PeopleCode Developer's Guide, "Creating Application Packages and Classes"*

---

## **Working with Definitions**

This section provides as overview of definitions and discusses how to:

- Open definitions.
- View definitions.
- View and edit definition properties.
- Create definitions.
- Save definitions.
- Close definitions.
- Rename definitions.
- Delete definitions.
- Find definitions.
- Use the Multiple Document interface.
- Use drag-and-drop mouse techniques.
- Use dynamic toolbars and menus.

- Use pop-up menus.
- Configure environments.
- Work with generated files.

## Understanding Definitions

You can create or modify many types of definitions in PeopleSoft Application Designer. This table lists development definitions:

<b>Definition</b>	<b>Description</b>
Activity	Workflow map showing the individual steps, events, and routings that comprise a complete activity in a business process.
Analytic Model	The representation of a methodology used exclusively with the PeopleSoft Analytic Calculation Engine. <i>See PeopleTools 8.52: Analytic Calculation Engine, "Designing and Editing Analytic Models."</i>
Application Engine	Program comprising SQL statements and PeopleCode programs that runs in batch mode as an alternative to using COBOL or SQR programs.
Application Package	Packages for creating Application Classes, which you can use to extend the functionality of existing PeopleTools classes (Rowset, Array, and so on) or to create your own classes.
Approval Rule Set	Workflow maps that provide a visual representation of approval rules. Virtual Approver reads approval rule sets to determine who must approve transactions.
Business Interlink	Gateway between PeopleSoft applications and the services of an external system.
Business Process	Workflow maps that provide a visual overview of the activities involved in a procedure.
Component	Representation of a logical business transaction or a set of logically related pages that are processed together.
Component Interface	Way of externalizing access to a component so that it can be used by a third party or an application message.

<b>Definition</b>	<b>Description</b>
Field	Individual pieces of data, such as an employee ID, that a user can enter and the system can store in the database, in a column on a table, or in a view.
File Layout	Definition (or mapping) of a file to be processed. It identifies where data fields are located in a file.
HTML	Definition that represents static HTML that can be included on a page definition.
Image	Definition that stores and displays graphics or other images, such as icons, that appear throughout the application.  <b>Note.</b> Image definitions and image fields are not the same.
Menu	Definition that organizes the components that you build and the pages contained in the components.
Message	Based on a multilevel structure, similar to components, a way of defining the data to insert into an application message at runtime.
Message Channel	Corresponding to groups of message definitions, these help order messages properly, enhance scalability, and provide a simple way to define processing characteristics of many similar messages as a single group.
Mobile Page	A PeopleSoft Pure Internet Architecture page based on a synchronizable component interface for display on disconnected mobile devices, such as laptops or PDAs.
Page	User-defined collection of controls created for entering, viewing, and editing data online. The system validates user input, writes it to the database, and retrieves and displays it upon request.
Project	User-defined collection of related definitions created for developing, adapting, or upgrading a PeopleSoft application.

<b>Definition</b>	<b>Description</b>
Problem Type	In optimization, a way to tie the optimization application records, the optimization transactions, and the optimization plug-in together as one entity.
Record	Tables that store all of the data that resides in PeopleSoft applications part of a relational database system. Tables are the same as records, and each record definition describes the properties of an underlying SQL table.
SQL	Entire SQL programs or fragments of SQL statements available for reuse.
Style Sheet	Collection of styles that PeopleSoft Pure Internet Architecture application pages can use.

## Opening Definitions

Use the definition workspace to access or create definitions.

To open a definition:

1. Select File, Open or click the corresponding toolbar button.

The Open Definition dialog box appears.

2. Select the type of definition that you want to open from the Definition drop-down list box.
3. Provide selection criteria.

Enter a definition name or description (or the beginning characters of either). Other selection criteria options might be available, depending on the definition type.

4. Click Open, or press ENTER, to display definitions matching the selection criteria that you entered.

To clear the current selection criteria and start over, click New Search. To change how the system displays the search list, perform one of these actions:

- Click the List button to see only the names of the definitions.
- Click the Details button to see the names and descriptions of the definitions. This is the default.

5. Select a definition to open.

6. Double-click the definition that you want to open in the definition workspace, or highlight the definition and click Open.

You can also use the shift-click selection technique to select more than one definition to open in a single action, or you can right-click to view a pop-up menu, where you can open, print, rename, or delete the selected definition.

---

**Note.** The print menu is disabled when you right-click any of the following definitions: activity, approval rule set, business process, component interface, component, project, and mobile page.

---

---

**Note.** When the definition is open and able to be modified, "Ready" appears in the lower left-hand corner of the workspace.

---

## Viewing Definitions

In the project workspace, you can view a definition that is referenced by the active definition.

To view a definition, select View, View Definition. You can also right-click the definition you want to view and select View Definition from the pop-up menu.


## Viewing and Editing Definition Properties

Properties are associated with each definition type. To view or edit definition properties, select File, Definition Properties. You can also click the toolbar button, press Alt+Enter, or right-click the definition and select Properties from the pop-up menu.

The Definition Properties dialog box almost always includes a General tab that provides a place to display and enter descriptions and comments about the definition. An optional Owner ID drop-down list box designates the owner of an individual definition. This ID helps identify which application team last made a change to a definition. Set the default owner ID for new definitions in the Options dialog box on the Owner ID tab.

**Record Properties**

General Use

 AMM\_DERIVED

Description: App Msg Monitor work record

Record Definition:

This derived record is used by the Application Message Monitor.

Owner ID: PeopleTools

**Last Updated**

Date/Time: 01/06/2011 9:26:53AM

By User: PPLSOFT

OK Cancel

Example of a Property sheet

With PeopleSoft Application Designer, every definition that you can open (menus, pages, components, records, fields, and so on) has an associated property sheet. Definition properties that are specific to the definition type are grouped together in these definition property sheets. For example:

- Record type and record use properties are found in the Definition Properties sheet for records.
- Translate values are found in the Definition Properties sheet for fields.
- Menu groups and menu labels are found in the Definition Properties sheet for menus.
- Search record, actions, and internet settings are found in the Definition Properties sheet for components.

## Creating Definitions

When building or adapting an application, try to use existing definitions as much as possible; when that is not possible, create new definitions.

To create a new definition:

1. Select File, New or click the corresponding toolbar button.
2. Select the definition type.

The new definition appears in the definition workspace.

## Saving Definitions

To retain your changes, you must save the definition. You must name new definitions before you can save them.

To save a definition:

1. Click the Save button on the toolbar or select either File, Save or File, Save As to save a definition.

The Save and Save As commands save a single definition. Use the File, Save All command or the Save All toolbar button to save all open definitions at once.

2. If you are saving a page or component, specify the language or market.

For a page definition, the Language drop-down list box appears; for a component, the Market drop-down list box appears.

### ***Definition Naming Conventions***

When naming definitions, use the following guidelines:

- Use only uppercase alphabetic characters (A–Z), numeric characters (0–9), or underscore (\_). Other special characters, such as the em dash (—), dollar sign, number sign, question mark, and so on, are not permitted.
- Names must begin with an alphabetic character.
- Embedded spaces are not permitted.

Instead, use an underscore.

- Avoid using accented characters.

Names containing these characters may not sort consistently, depending on workstation or locale settings, which results in problems during an upgrade.

- System definitions such as records, fields, styles, and so on generally have a name and a label or description. Generally, you should use US-ASCII characters only for the definition name. Labels or descriptions, which appear to the user, can contain non-ASCII characters.

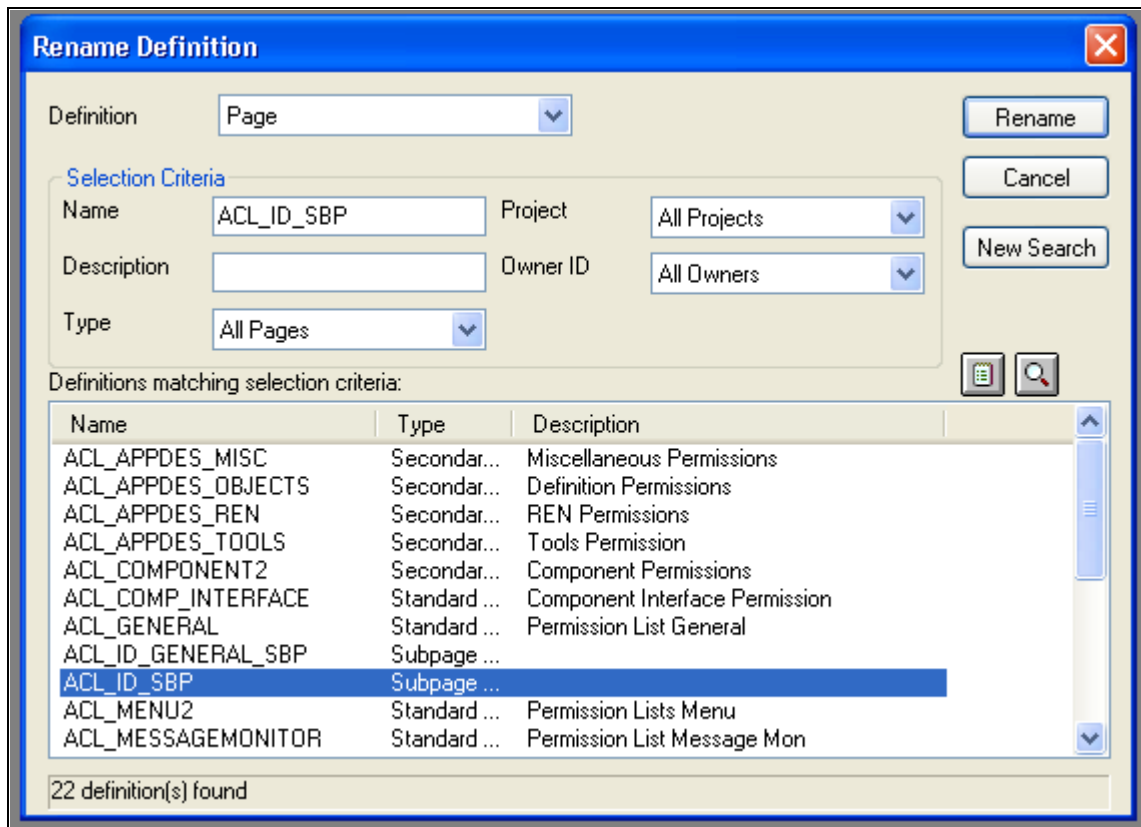
**Note.** Existing Business Process and Business Process Map definitions may already contain mixed case characters. Nonetheless, newly created definition names must be all uppercase.

## Closing Definitions

To close a definition, either select File, Close or click the Close Window icon on the title bar of the window of the active definition. You can also select Window, Close All to close all open definitions.

## Renaming Definitions

You can rename existing definitions in a database at any time.



Rename Definition dialog box

To rename a definition in the database:

1. Select File, Rename.

The Rename Definition dialog box looks like the Open Definition dialog box, and you select definitions in the same way.



2. Specify the definition type and any selection criteria that are useful in locating the definition that you want to rename.

The Rename Definition dialog box does not have a Select button. So, after entering either a partial name or description of the definition, press Enter to retrieve a list of definitions matching your selection criteria.

3. Select the definition that you want to rename.
4. Click the Rename button in the dialog box and double-click the selected definition, or right-click to select Rename from the pop-up menu.

The selected definition name in the Rename Definition dialog box is in edit mode.

5. Change the selected definition name and press Enter.

When you press Enter, PeopleSoft Application Designer prompts you first, to close any open definitions before you can save the new name, and then to confirm the rename before you can save a change.

6. Click OK to complete the renaming of the selected definition.

## Deleting Definitions

Deleting a definition is different from removing a definition from a project. When you remove a definition from a project, the reference to the definition is removed from the project, but the definition still exists in the database and, thus, may be part of another project. Deleting a definition, however, permanently removes the definition from the database. Any projects or other definitions that contain the deleted definition are affected.

Before deleting a definition you should find all references to that definition by running the Find Definition References process. For example, a field definition that you want to delete may be referenced by both record and page definitions.

---

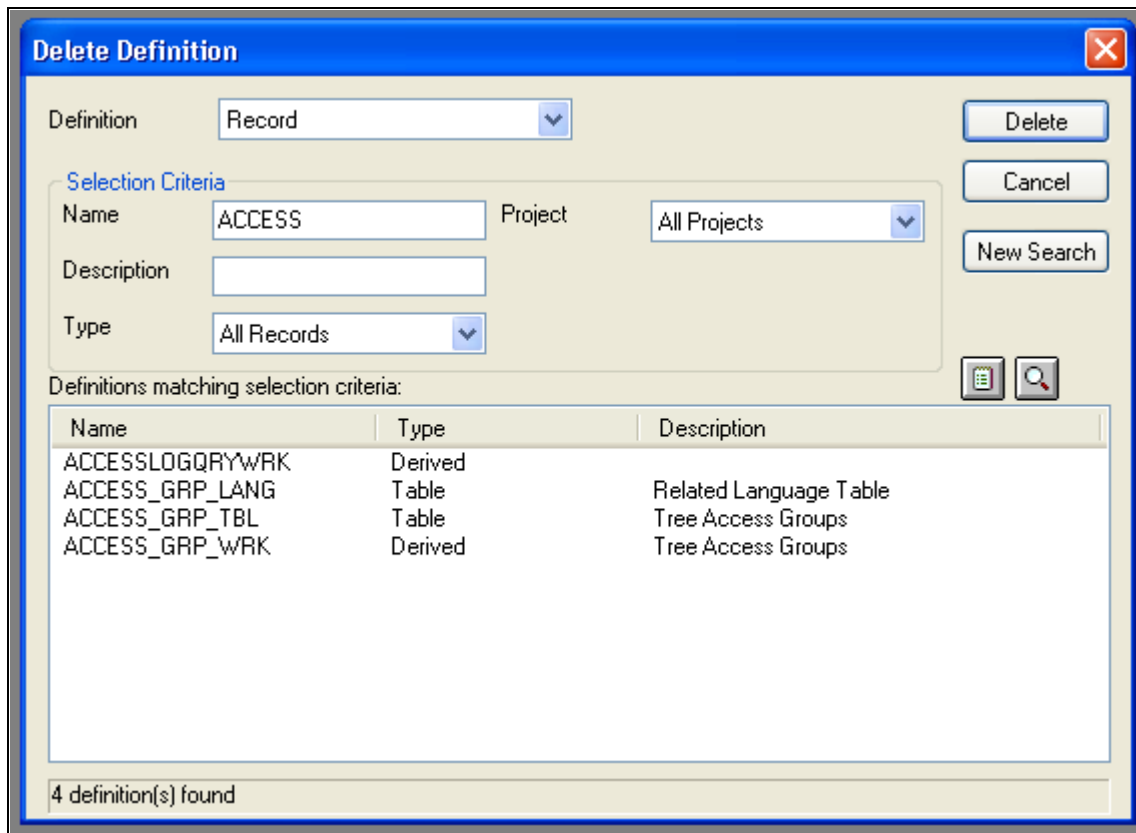
**Note.** If you delete a component definition, you must delete all component interfaces that are based on that definition.

---

---

**Note.** A component definition must contain at least one page. If you attempt to delete a page definition that is the only page definition within a component, the system does not allow the delete to occur and displays an error message. If the page definition is the only page within a component, the error message appears during a File, Delete action or during an upgrade copy to database (with a Delete action).

---



Delete Definition dialog box

To delete a definition from the database:

1. Select File, Delete.

The Delete Definition dialog box looks like the Open Definition dialog box, and you select definitions in exactly the same way.

2. Specify the definition type and any selection criteria that are useful in locating the definition that you want to delete.

After entering either a partial name or description of the definition, press Enter to retrieve a list of definitions matching your selection criteria.

3. Select the definition that you want to delete.

To select multiple definitions, hold down either Shift or Ctrl while clicking the appropriate definitions.

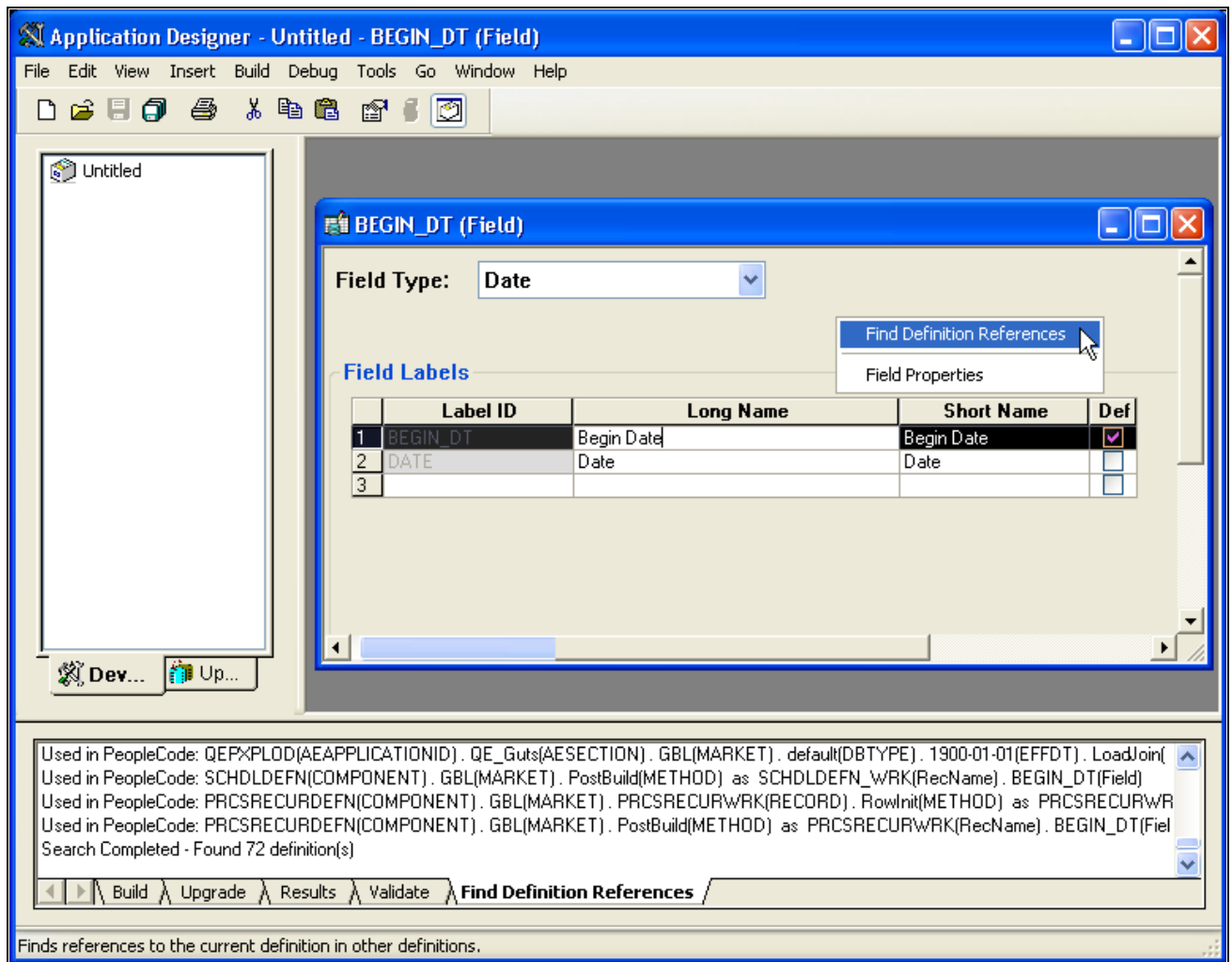
4. Click the Delete button and confirm that you want to delete the selected definition.

## Finding Definitions

Before making significant changes, such as deleting a definition, you must see what other parts of the application might be affected by the change. The Find Definition References feature performs a comprehensive search of the database pinpointing all definitions in which the changed definition is located.

Although a project keeps track of your definitions, the definitions are not actually embedded in the project.

**Note.** The distinction between *related* definitions and *referenced* definitions is significant. A related definition is one that *is used by* the current definition. A referenced definition is one that *uses* the current definition. For example, in the case of a record definition, fields in the record are related definitions. The records that use the fields are its definition references.



### Finding definition references

To find definition references:

1. Open the definition in the definition workspace.
2. Select Edit, Find Definition References, or right-click the definition and select Find Definition References from the pop-up menu.

After you select this item, a search of the database takes place, and the results appear on the Find Definition References tab of the output window.

3. Select any definition that appears in the output window by double-clicking it.

The selected definition opens in the definition workspace.

4. After you find a definition, select the definition reference and right-click to view the pop-up menu.

---

**Note.** Double-clicking an entry in the Find Definition References output window opens the appropriate mode of PeopleSoft Application Designer. If you double-click a related record that was found, the record opens and the field for which you were looking is highlighted.

---

## Using the Multiple Document Interface

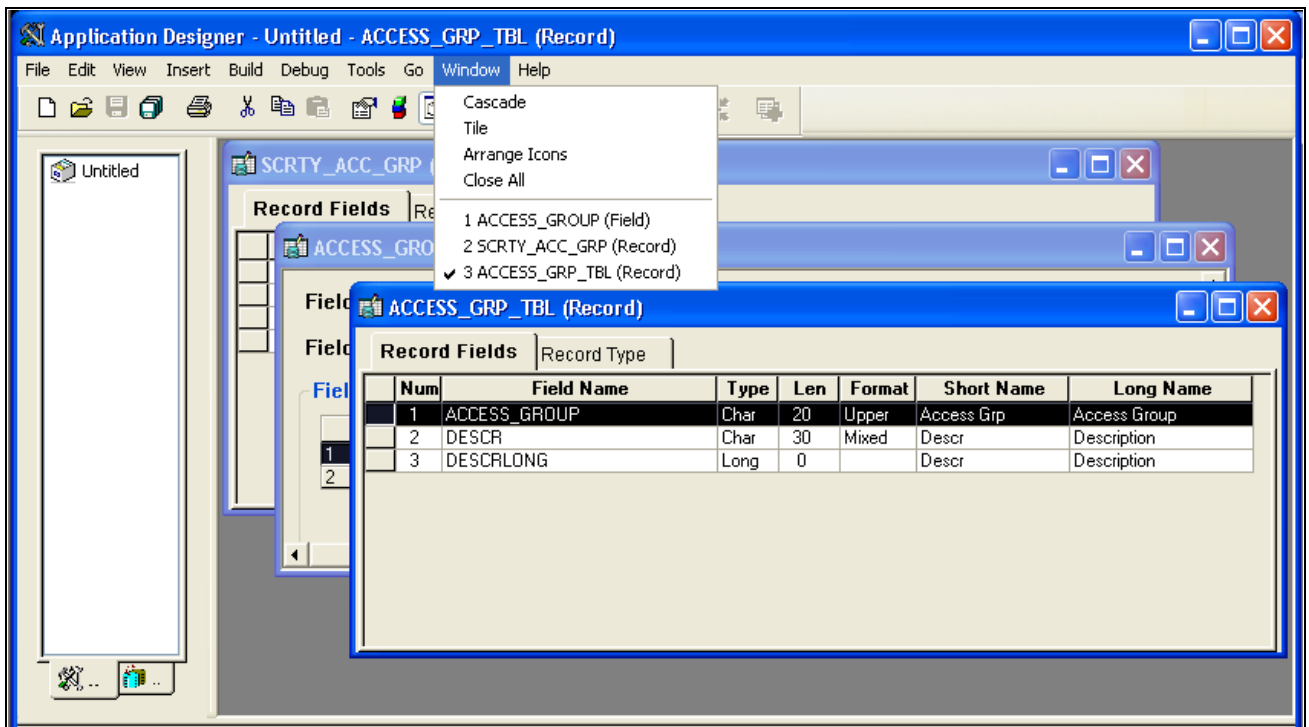
PeopleSoft Application Designer makes use of multiple document interface (MDI). Consequently, each window in the definition workspace of PeopleSoft Application Designer represents a view of a definition, and you can have multiple windows open at the same time. These definition windows must stay inside the borders of the definition workspace, but you can maximize, minimize, cascade, or tile them in that space.

You can have different types of definition windows open at the same time, such as menu, field, record, and page windows. As a result, you can work with different definitions simultaneously, which simplifies the process of designing a group of related definitions.

### **Active Windows**

Although you can have multiple windows open at the same time, only one window can be active at any moment. Make a window active by selecting it in the definition workspace. The active window always appears on top with the title bar in *active window title* color.

In addition to clicking a window to make it active, you can also use the Window menu shown in the following example. This menu displays the windows that are currently open by displaying a check mark next to the active one. Change the window that is currently active by selecting it from the menu.



Example of the Window menu

Other commands from the Window menu include Cascade, Tile, Arrange Icons, and Close All windows.

## Using Drag-and-Drop Mouse Techniques

PeopleSoft Application Designer makes ample use of drag-and-drop mouse techniques. For example, you can drag a field from a record definition and drop it onto a page definition. You can then drag the page onto a component and drag the component onto a menu. Using the drag-and-drop technique is faster and simpler than using menu commands to manipulate definitions.

To drag and drop a definition:

1. Open the definition in either the project workspace or the definition workspace from which you want to drag.

For example, to drag a field from an existing record definition to a new page definition, first open the record containing the field. You can open it in the definition workspace or expand the record in the project workspace so that the fields appear. With the appropriate field displayed, you can drag and drop it into another definition.

2. Open the definition to which you want to drag.
3. Drag the definition from one definition to the other.

## Using Dynamic Toolbars and Menus

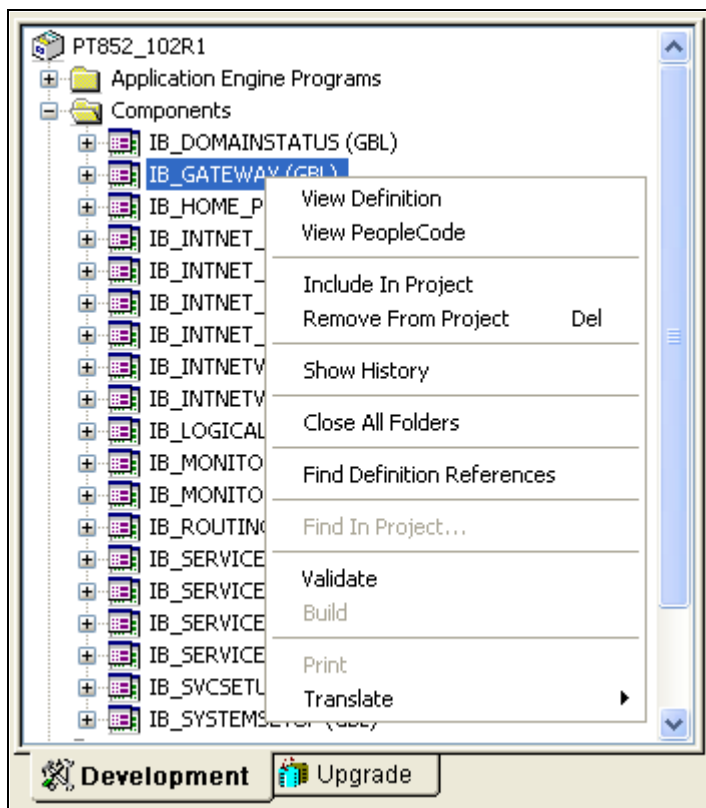
The PeopleSoft Application Designer toolbar and menus change based on the type of definition that is active. For example, when a page definition is active, the toolbar displays buttons that represent the definitions that you can add to a page. This technique is also known as *morphing*.

The change is so subtle that you can easily miss it. The change is easy to miss because the menu bar names are consistent among definition types. However, if you look at the menu items and toolbars closely, you see a distinct difference. For example, if a page window is active, then the menus and toolbars change to include actions and options applicable to a page definition, as in the Show Grid option on the View menu and the introduction of the Layout menu. In addition, page-specific toolbars appear.

In the same respect, if a record window is active, the menus and toolbars morph to include actions and options applicable to a record definition. For example, different options (such as View, Field Display, and so on) are available in the View menu. Different toolbar buttons for record definition actions also appear.

## Using Pop-up Menus

Pop-up menus are context-sensitive menus that appear at the current location of the mouse pointer whenever you click the alternate mouse button (typically, the right mouse button). They provide efficient access to numerous PeopleSoft Application Designer commands. Because pop-up menus appear at the current location of the pointer, they eliminate the need to move the pointer up to the menu bar or to a toolbar. A pop-up menu contains only the commands that apply to the selected definition or the current context. This example shows the component definition pop-up menu:



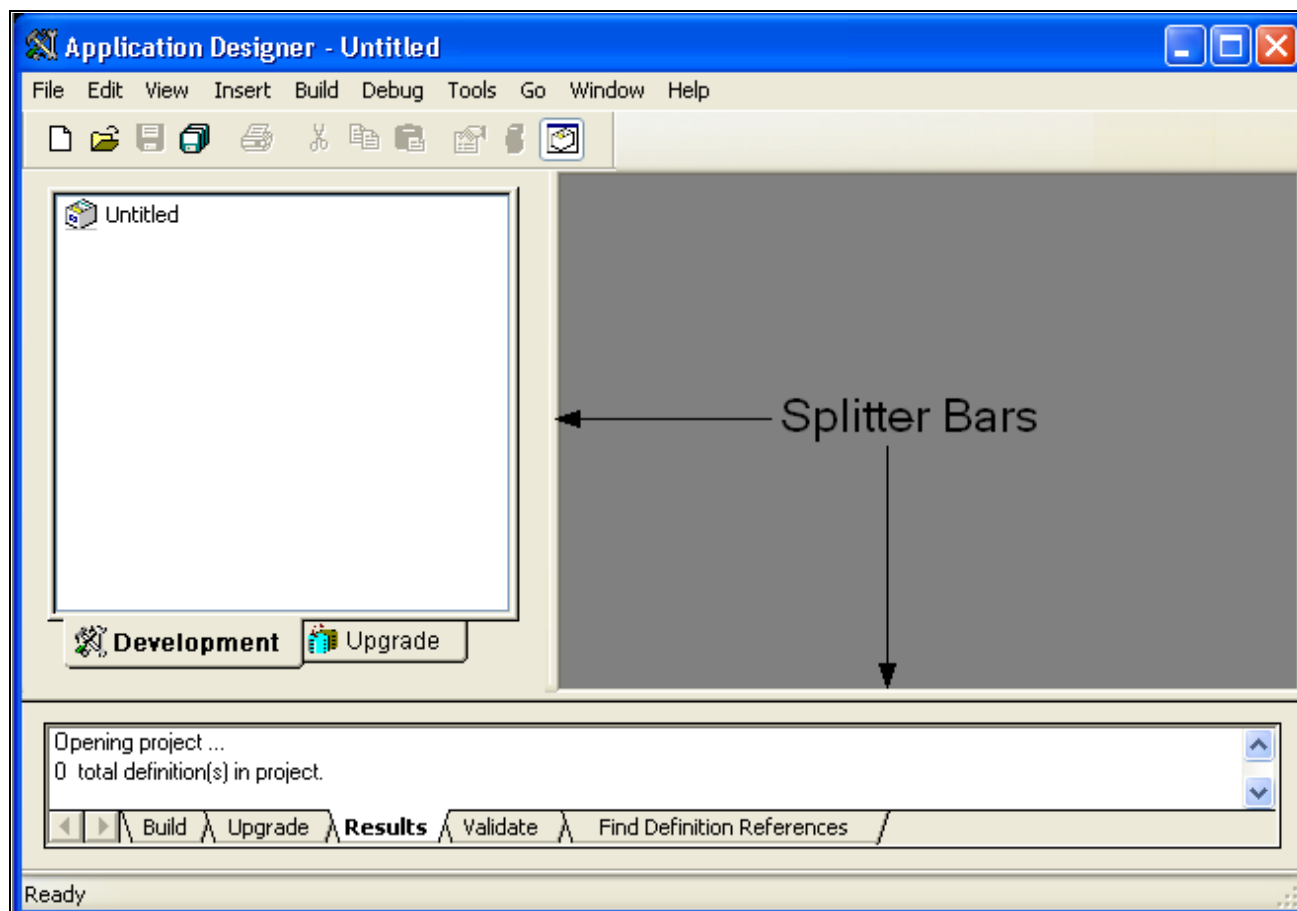
Example of the Component definition pop-up menu

## Configuring Environments

PeopleSoft Application Designer is highly configurable, enabling you to tailor the tool to your needs or preferences.

### ***Splitter Bars***

A splitter bar is a thin vertical or horizontal bar that separates windows inside a parent window. Using the splitter bar enables you to resize two windows at the same time. Splitter bars are used between the project workspace, definition workspace, and output window. This example shows splitter bars:



Example of Splitter bars used to resize a workspace

Drag the splitter bar to change the size of the project workspace, definition workspace, or output window.

### ***Dockable Windows and Toolbars***

You can freely move PeopleSoft Application Designer windows and toolbars to either anchor (dock) to parts of the main application window or float freely anywhere that you want them to appear. When you dock a toolbar or window, you anchor it to the top, sides, or bottom of the main window.

To move or dock a PeopleSoft Application Designer window or toolbar, drag the window title bar or the appropriate toolbar to a new location. If you move the window or toolbar near the top, bottom, or sides of the main window, the window or toolbar snaps into place, meaning that it is docked. However, if you press and hold Ctrl and then drag the window or toolbar, you can prevent it from automatically docking when it is near the edge of the window.

---

**Note.** Use caution when running the PeopleCode Debugger with undocked windows.

---

### ***View Menu***

The View menu enables you to display or hide different PeopleSoft Application Designer elements, such as the project workspace, output window, and toolbar.

## **Working with Generated Files**

When working with PeopleSoft Application Designer, you often generate scripts, logs, reports, and so on. When specifying the full path for a file reference in PeopleSoft Application Designer, you are limited to 100 characters. This limit applies to all file references, which include build scripts, build logs, compare reports, and so on. For example, a reference to a build log, such as `c:\temp\...\psbuild.log`, cannot exceed 100 characters.

---

## **Using Miscellaneous Definitions**

Several definition types differ from the standard definitions described in previous sections. These definitions control your work environment or affect other definitions that you create. These miscellaneous definitions are found under the Tools menu and include:

- Field Formats
- System Colors
- Styles
- Access Profiles
- File Type Codes

### ***Field Formats***

PeopleTools provides formatting at both the field and page levels. Field format specifications affect internal values that are stored in the database, and page format specifications affect the visual presentation of values. Several formatting options are available for U.S. Social Security Numbers, Canadian Social Insurance Numbers, postal codes, telephone numbers, and various currencies, dates, and times.

Custom field formats extend these formatting provisions to support the same types of data in other countries. They also support other types of formatted information, such as bank codes, credit card numbers, part numbers, and serial numbers.

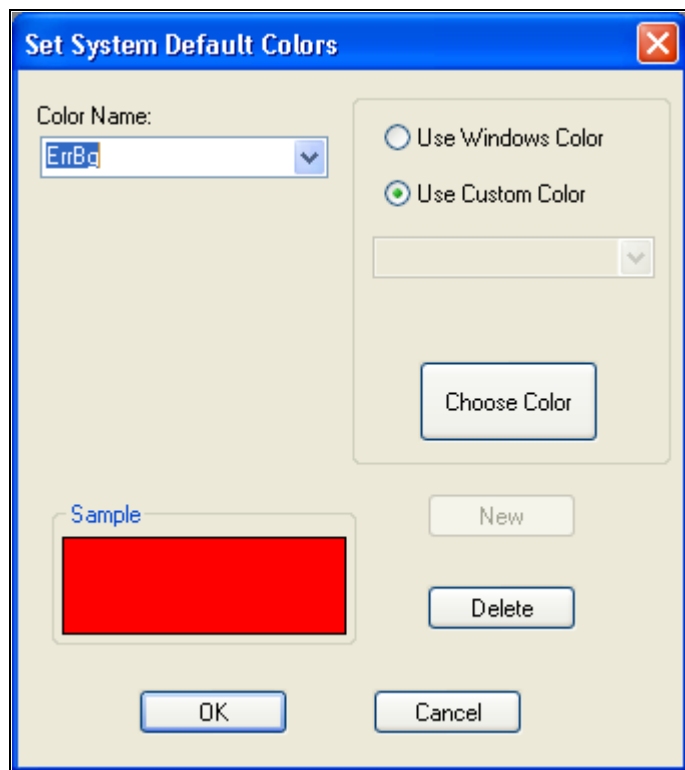
Custom field formats enable you to create your own format definitions using format notation and apply them to fields. These formats are organized into format families, which can include one or more unique formats.



See [Chapter 4, "Creating Field Definitions," Changing Format Families, page 86.](#)

## System Colors

You can establish definitions for colors that you use in your applications. You define colors by associating a color name either with an existing color in the PeopleSoft Internet Architecture or with a custom color that you define. Since taste in color is subjective, PeopleTools allows individual operators to customize their color settings. Users can use the default color definitions you establish in PeopleSoft Application Designer, or they can override them with their own color definitions.



Setting default system colors

To modify default system colors:

1. Select Tools, Miscellaneous Definitions, System Colors.
2. Select the Color Name you want to modify.
3. To change the color, click the Choose Color button.
4. Choose from the available colors or click Define Custom Colors to create your own color.
5. Click OK.

The new color appears in the Sample area of the Set System Default Colors dialog box.

To create new system colors:

1. Type a new name in the Color Name field and click the New button.
2. Select the Use Custom Color radio button to select a custom color.

3. Click the Choose Color button.
4. Choose from the available colors or click the Define Custom Colors button.
5. Click OK in both dialog boxes to accept the new color.

## **Styles**

This feature is deprecated. You set styles using style sheet definitions.

See [Chapter 15, "Creating Style Sheet Definitions," Understanding Style Sheets and Style Classes, page 445.](#)

## **Access Profiles**

Access profiles are the security profiles the system uses to connect to the PeopleSoft database. The *PeopleTools 8.52 PeopleBook: Security Administration* provides detailed information about access profiles.

See *PeopleTools 8.52: Security Administration*, "Administering User Profiles," Setting Up Access Profiles.

## **File Type Codes**

You may need to set up file type codes when creating change projects. You create change projects to work with the PeopleSoft Application Designer Change Packager tool to package system customizations in preparation for an upgrade. The file type code stores generic information that is applicable to a group of files within the same target directory.

See *PeopleTools 8.52: Change Assistant*, "Working with Change Packages."

---

# **Using the PeopleSoft Application Designer Spell Check Feature**

This section provides an overview of the PeopleSoft Application Designer spell check feature and discusses how to:

- Enable the spell check feature.
- Configure browser options.

## **Understanding the Spell Check Feature**

The PeopleSoft Application Designer spell check feature checks the spelling for user-facing labels and text in projects and legacy data. Use this feature before building your application. The spell check feature verifies spelling for these items:

- Message Catalog entries.
- Page definitions (label text).
- XLAT tables.
- Component definitions (tab and menu labels).

- Portal registry (labels and attribute labels).
- Record field labels.

### **Spell Check Results**

The system logs spell check results on the Spell Check tab in the PeopleSoft Application Designer output window. The text *Begin Spell Check* always appears as the first line on the tab. The first line for each spell check category indicates the total number of entries that were checked for the project. The last line in the output window indicates the total misspelled words encountered during the check.

For each spell check category the results may read differently. This table describes how to read the spell check results for each category:

<b>Output by Error Type</b>	<b>Description</b>
Message Catalog: Message Set www Nbr xxx (yyy) - zzz For example: Message Set 1 Nbr 1 (Message Text) - Distributed Object Mgr: Xelp Xme=Xangage=abc rtz	Where: www = Message set number. xxx = Message number within the message set. yyy = : <ul style="list-style-type: none"> <li>• Message text, indicating that zzz is the complete message text.</li> <li>• Explain text, indicating that zzz is the complete explain text.</li> <li>• Misspelling, indicating that zzz is a misspelled word in either message text or explain text.</li> </ul>
XLAT: [vvv.www.xxx] (yyy) - zzz	Where: vvv = Database field name. www = Field value. xxx = Effective date. yyy =: <ul style="list-style-type: none"> <li>• Long name, indicating that zzz is the complete text for the long name.</li> <li>• Short name, indicating that zzz is the complete text for the short name</li> <li>• Misspelling, indicating that zzz is a misspelled word in either a long or short name.</li> </ul>

<b>Output by Error Type</b>	<b>Description</b>
<p>Component:</p> <p>[vvv.www] Item Name xxx (yyy) - zzz</p>	<p>Where:</p> <p>vvv = Component name.</p> <p>www = Market code.</p> <p>xxx = Item name for the component.</p> <p>yyy =:</p> <ul style="list-style-type: none"> <li>Item label, indicating that zzz is the complete text for the item label of the component.</li> <li>Folder tab label, indicating that zzz is the complete text for the folder tab label of the component.</li> <li>Misspelling, indicating that zzz is a misspelled word in either the item label or the folder tab label.</li> </ul>
<p>Portal Registry:</p> <p>[ttt.uuu.vvv] www xxx (yyy) - zzz</p>	<p>Where:</p> <p>ttt = Portal name.</p> <p>uuu = C for content reference, F for folder.</p> <p>vvv = Portal object name.</p> <p>www =:</p> <ul style="list-style-type: none"> <li>Empty.</li> <li>Attribute.</li> <li>System attribute.</li> </ul> <p>xxx =:</p> <ul style="list-style-type: none"> <li>Empty if www is empty.</li> <li>Attribute name if www is attribute.</li> <li>System attribute name if www is system attribute.</li> </ul> <p>yyy =:</p> <ul style="list-style-type: none"> <li>Label name, indicating that zzz is the complete text for the label name.</li> <li>Display name, indicating that zzz is the complete text for the display name of an attribute.</li> <li>Misspelling, indicating that zzz is a misspelled word in either the label name or the display name.</li> </ul>

<b>Output by Error Type</b>	<b>Description</b>
Field Label: [www.xxx] (yyy) - zzz	Where: www = Field name. xxx = Field label name. yyy = <ul style="list-style-type: none"> <li>• Long name, indicating that zzz is the complete text for the long name.</li> <li>• Short name, indicating that zzz is the complete text for the short name</li> <li>• Misspelling, indicating that zzz is a misspelled word in either a long or short name.</li> </ul>
Page: [www] Field ID xxx (yyy) - zzz	Where: www = Page name. xxx = Field ID within the page. yyy = <ul style="list-style-type: none"> <li>• Label text, indicating that zzz is the complete text for the label text of the field.</li> <li>• Scroll area, indicating that zzz is the complete text for label text within a scroll area.</li> <li>• Misspelling, indicating that zzz is a misspelled word in the label text.</li> </ul>

You can launch an identified definition directly from the output window by double-clicking the error line. PeopleSoft Application Designer launches the appropriate content to correct the misspelling, as described in this table:

<b>Error Type</b>	<b>Launches</b>
Message Catalog	NA
XLAT	Field definition.
Component	Component definition.
Portal registry	Browser to display the Content or Folder Administration page.
Field label	Field definition.
Page	Page definition.

## Enabling the Spell Check Feature

The spell check feature is turned off by default. You enable the spell check feature in the Options dialog box on the Validate tab. Enabling the spell check option ensures that spell check is run on the designated project when you validate the project.

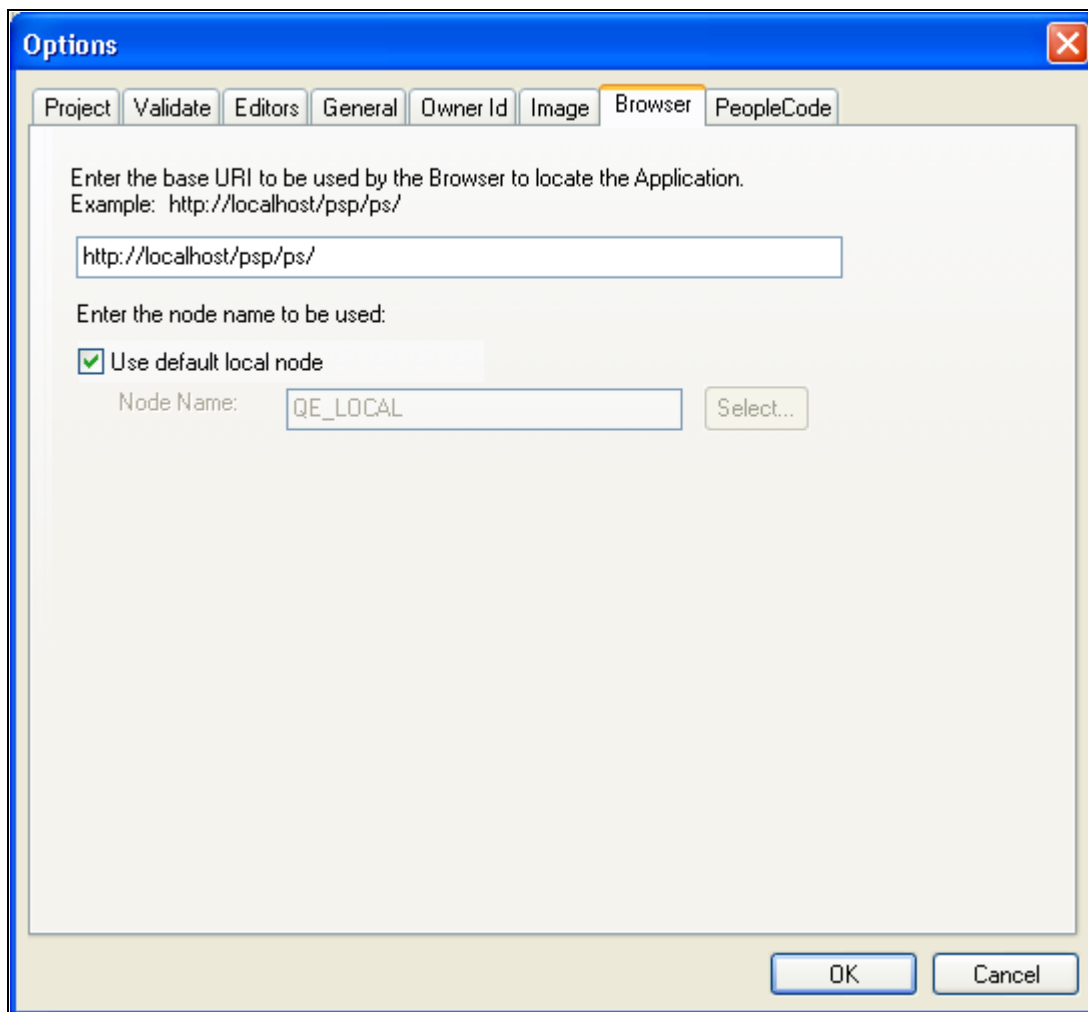
To enable spell check in PeopleSoft Application Designer:

1. Select Tools, Options from the PeopleSoft Application Designer menu.
2. Select the Validate tab.
3. Select the Validate Spelling check box.
4. (Optional) Select the Log to file check box if you want the spell check results to be saved to a file on your system.
5. Enter a file name in the File name field or click the browse button to search for a specific file.

Otherwise, the system applies the default file name of *spellcheck.log* and saves the results to the temp directory. This field is required when the Log to file option is selected.

## Configuring Browser Options

The Browser tab on the Options dialog box enables you to specify values that are required to compose a URL for the browser to locate a portal page. The spell check feature uses the information recorded in this dialog box to launch portal pages from the Spell Check tab in the output window when you double-click a portal registry misspelling.



Options dialog box: Browser tab

To configure browser options:

1. Select Tools, Options and click the Browser tab.
2. Enter the base URI or use the default setting.

This value is the URI of the node that hosts the portal.

3. Set the node name to be used with the base URI.

The Use default local node check box is selected by default. Deselecting this check box enables the Select button, which allows you to set the node name in the Select Node dialog box.

The Select Node dialog box displays the list of available nodes by project. You can search for the appropriate node by clicking the Select button.

---

## Building and Maintaining Data

This section discusses how to:

- Build and maintain data.
- Create SQL tables.

## Building and Maintaining Data

You must keep underlying database definitions synchronized with PeopleSoft definitions. Therefore, from PeopleSoft Application Designer, you must:

- Submit SQL Alter and Create commands for all types of database definitions.
- Manage the database index.
- Manage the table space.
- Manage the DDL model for tables, indexes, and table spaces.

## Creating SQL Tables

To access data, you must create a SQL table and then store data in the fields and records. The process of running the SQL to synchronize the database with records, indexes, and DDL is called the Build process.

The Build feature processes definitions at these levels:

- Current record.
- Selected records in the project workspace.
- All records in the project.

### **See Also**

Chapter 7, "Administering Data," page 155



## Chapter 3

# Working With Projects

This chapter provides an overview of projects and discusses how to:

- View projects.
- Create and save projects.
- Set project properties.
- Insert definitions into projects.
- Remove definitions from projects.
- Merge projects.
- Use change projects.
- Set project options.
- Validate projects.
- Set system IDs.

---

## Understanding Projects

While not one of the eight steps of application development, using projects is an efficient way to organize your definitions as you develop and configure your application. A project keeps track of all definition types as a simple list of definition names; the definitions are *not* stored in the project. Development definitions exist outside of the project in your PeopleSoft database.

You are not required to use projects in PeopleSoft Application Designer; and you can create and edit individual definitions without associating them with a project. However, using projects can help you:

- Organize related definitions.
- Understand relationships among definitions.
- Coordinate the work of several developers.
- Streamline upgrade tasks.
- Search for fields or records.
- Access related definitions easily.

## Viewing Projects

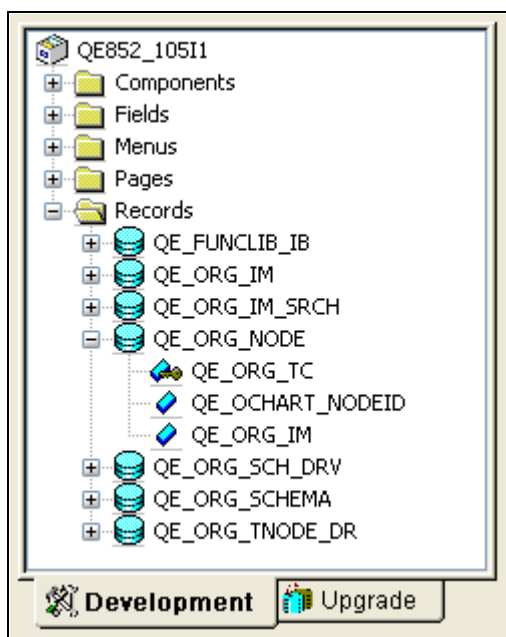
On the left-hand side of your screen, the project workspace displays one project at a time and all of its definitions, making them easily accessible for opening and editing. The project workspace includes two views representing the collection of definitions in the project: Development and Upgrade. Switch between the views by selecting the folder tabs at the bottom of the project workspace.

This section discusses how to:

- Access project definitions.
- Use the upgrade view.

## Accessing Project Definitions

In the Development view, the project folder contains a folder for each definition type included in the project. To open folders and view related definitions, click the expand (+) button. Access project definitions by double-clicking the definition. This example shows the Development view in the project workspace:

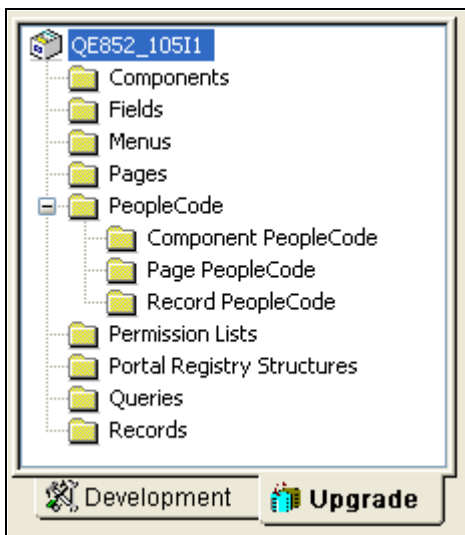


Example of the Development view in the project workspace

The definitions are organized by definition type and appear in a project tree. The project tree enables you to see all definitions associated with the project and their connection by showing the definitions in an hierarchical format. By expanding definitions that have nested levels, such as records, you can see the field definitions included in that record.

## Using the Upgrade View

The Upgrade view helps to streamline the migration of definitions—such as records, pages, or PeopleCode—from one PeopleSoft database to another. It displays all of the definitions available for upgrade and the attributes related to the upgrade process. When you double-click a definition type in the Upgrade view, an upgrade definition window appears in the definition workspace, displaying the definitions of that type that are available for upgrade and the associated upgrade options. This example shows the Upgrade view in the project workspace:



Example of the Upgrade view in the project workspace

The Upgrade view is important when you move changes into production and when you upgrade to new PeopleSoft releases. Definitions in development projects and upgrade projects are interchangeable, meaning that users can use the Upgrade view to copy or compare components that were developed in PeopleSoft Application Designer.

### See Also

*PeopleTools 8.52: PeopleSoft Application Designer Lifecycle Management Guide, "Upgrading with PeopleSoft Application Designer"*

---

## Creating and Saving Projects

This section discusses how to:

- Create new projects.
- Save projects.

---

**Note.** Changes that you make to a definition in a project are also reflected outside the project. Before you change a definition in a project, be aware of how it might affect other definitions outside the project. For example, when you rename or delete a field, you do so globally. Although you can enter a project name as selection criteria in certain PeopleSoft Application Designer dialog boxes (such as Open, Rename, and Delete), this is just to narrow the list of definitions returned in the selection list.

---

## Creating New Projects

When you start PeopleSoft Application Designer, an empty project titled *Untitled* always appears, whether you use the project or not. Unlike other definitions, only one project can be open at a time. Although you are encouraged to develop in projects, you can hide the project workspace window and ignore the project entirely.

To create a new project:

1. Select File, New.
2. Highlight Project as the new definition type to create and click OK.

---

**Note.** Another way to create a new project is to open an existing project, select File, Save Project As, and then enter a new name for the project. The system copies all of the existing definitions in the project.

---

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33](#).

To open an existing project, follow the same procedure as you would for any other definition type.

See [Chapter 2, "Using PeopleSoft Application Designer," Opening Definitions, page 30](#).

## Saving Projects

When you save an entire project, you use a different command than what you use to save other types of definitions.

To save a project, select File, Save Project. You can also select File, Save Project As to save the project with a new name. Selecting File, Save Project As is a good way to make a copy of a project because the original project still exists under the original name.

The Save toolbar button saves whichever definition is active in the definition workspace, but you cannot use it to save an entire project.

---

## Setting Project Properties

Use the Project Properties dialog box to enter information about a project. Access Project Properties from the File menu.

**Project Properties**

General | Update IDs | Pre-Requisites

PPLTLS85

Description: PeopleTools 8.50 Project

Comments: PeopleTools 8.50 Project

☒ Change Project

Owner ID: ▼

**Last Updated**

Date/Time: 05/11/01 12:13:52AM

By User: QEADMIN

**Application Upgrade Target**

Server:

Database:

User:

OK Cancel

Project Properties dialog box

On the General tab, enter a description of the project and any additional comments you might want others to be aware of when using the project. Select the Change Project check box only if you are creating a change project to be used with the Change Packager. Selecting this option enables the Update IDs and the Pre-Requisites tabs, which also are used only with the Change Packager.

See *PeopleTools 8.52: Change Assistant*, "Working with Change Packages."

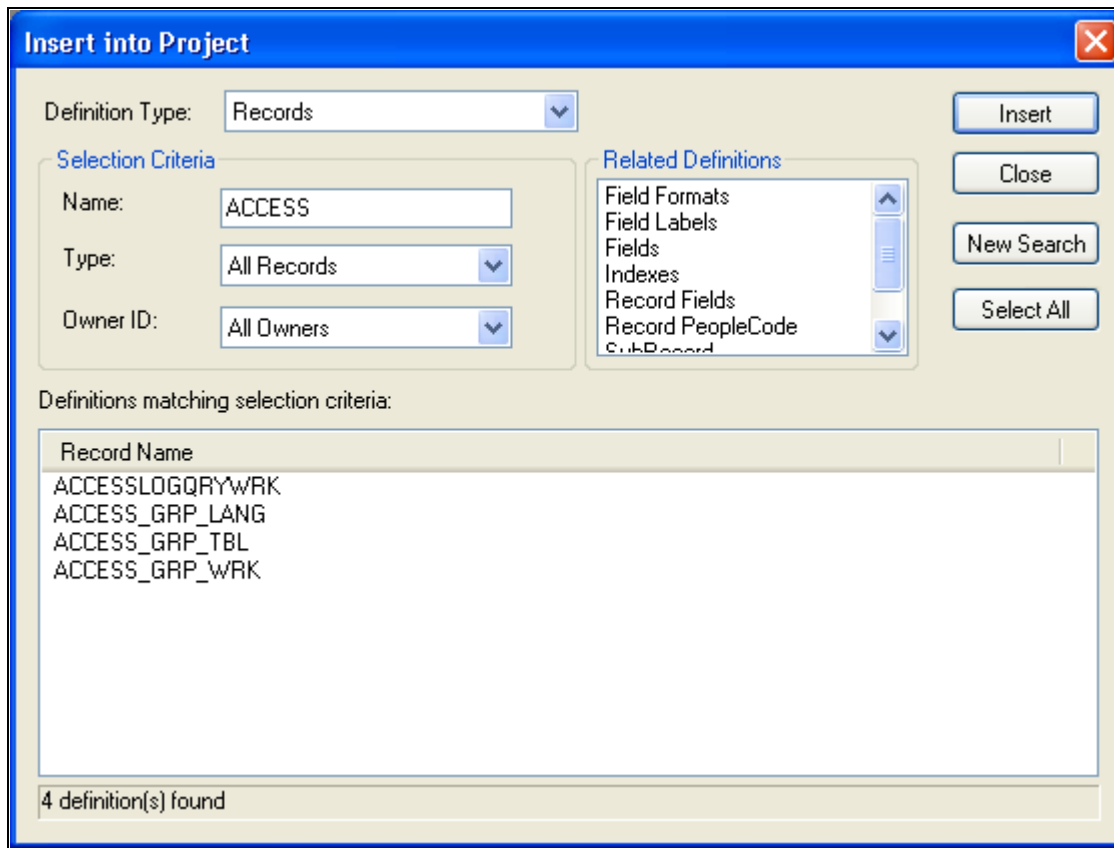
Select the appropriate Owner ID for the project. The Owner ID allows for a way to identify which definitions are owned by which PeopleSoft applications, such as PeopleSoft General Ledger, Accounts Receivables, and so on. The values in the drop-down list box are translate table values associated with the OBJECTOWNERID field.

## Inserting Definitions Into Projects

You can add definitions to a project by selecting Definitions into Project from the Insert menu.

This section discusses how to:

- Insert individual definitions into a project.
- Insert a group of definitions into a project.



Insert into Project dialog box

## Inserting Individual Definitions into a Project

To insert individual definitions into a project:

1. Open the definition that you want to insert into the project.

Make sure that the definition is the active definition in the definition workspace.

2. Select Insert, Current Definition into Project.

The system adds the definition to whichever project is currently open.

## Inserting a Group of Definitions into a Project

To insert a group of definitions into a project:

1. Select Insert, Definitions into Project.

The Insert into Project dialog box appears.

2. Select the type of definition to insert from the Definition drop-down list box.

To see all available definitions of one type, including upgrade-only definition types, click the Insert button or press Enter.

3. (Optional) Enter the selection criteria.

<b>Name</b>	Enter the definition name or a partial field name and press ENTER.
<b>Type</b>	Narrow your search by selecting a specific definition type, such as View for a record definition.
<b>Owner ID</b>	Narrow your search further by selecting an application with which the definition is used.

4. Select the definitions to insert.

To select multiple definitions, hold down either the Shift key or the Ctrl key while clicking the appropriate definitions.

5. (Optional) Specify which related definitions to insert.

To insert related definitions, specify them in the list of related definitions. To select multiple related definitions, hold down either the Shift key or the Ctrl key while clicking the appropriate definitions.

---

**Note.** After you insert definitions into a project, the upgrade-only definitions appear in the Upgrade view of the project workspace, not the Development view.

---

The status bar at the bottom of the Insert into Project dialog box indicates that definitions were inserted. Additionally, the Results tab on the output window displays the number of definitions inserted each time you perform an insert.

---

## Removing Definitions From Projects

Removing a definition from a project is different than deleting it from the database. When you remove a definition from a project, the reference to the definition is removed from the project, but the definition still exists in the database and thus may be part of another project.

To remove definitions from projects:

1. In the project workspace, select the definition that you want to remove from the project.

To select multiple items, highlight one or more nodes by using the Ctrl key while clicking the definitions to remove.

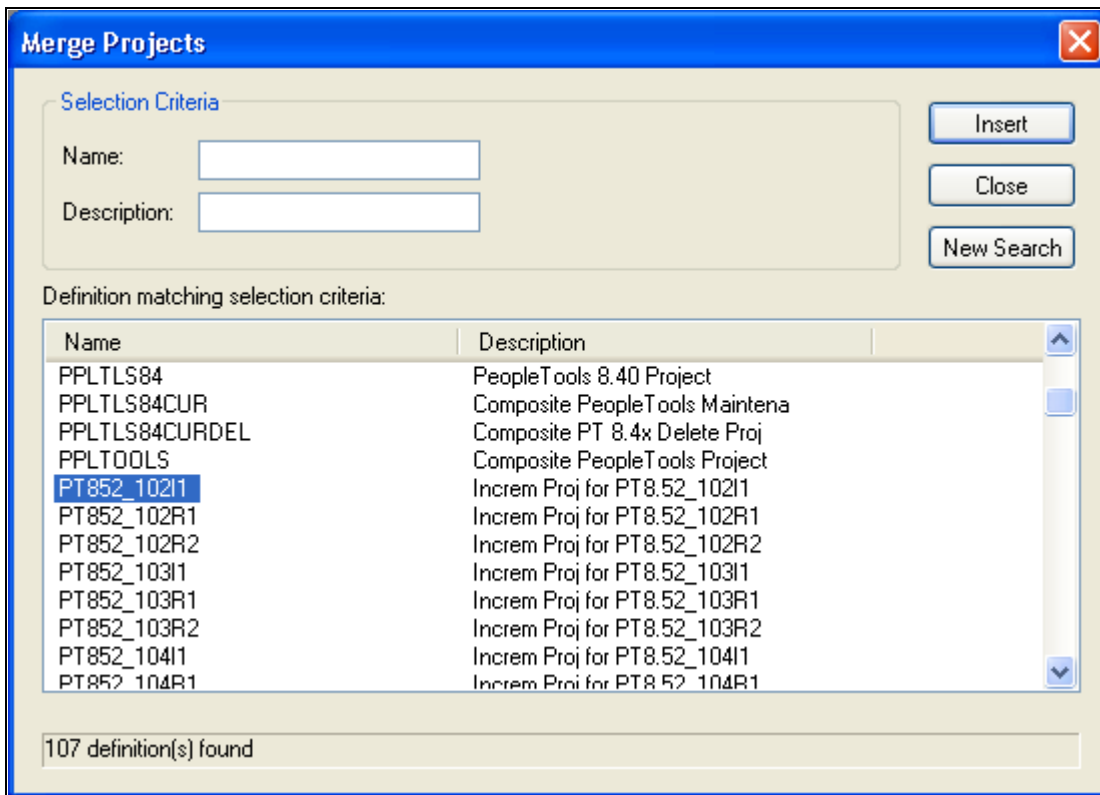
2. Press the Delete key or right-click to select Remove from Project from the pop-up menu.

This action does not delete the definition; it only removes it from the project.

---

## Merging Projects

You can merge two or more projects by inserting all of the definitions from one project into another.



Merge Projects dialog box

To merge projects:

1. Open the project in which you want to insert another project.
2. Select File, Merge Projects.  
The Merge Projects dialog box appears.
3. Enter selection criteria.
  - a. Enter a project name or description (or the beginning characters of either).
  - b. Click Insert or press the Enter key to display projects that match the selection criteria you entered.

4. Select a project to insert into your currently open project.

Double-click the project that you want to insert or highlight the project and click Insert. To select multiple projects, use the Shift and Ctrl keys.

5. Click Insert.

The system inserts all of the selected definitions into the project in one action. After each insert, note the information in the status bar and on the Results tab in the output window.



---

## Using Change Projects

Use change projects with the Change Packager feature to group definitions that are affected by PeopleSoft application updates. Combined with the PeopleSoft Change Assistant, the Change Packager feature significantly reduces the complexity and time required to apply PeopleSoft application updates.

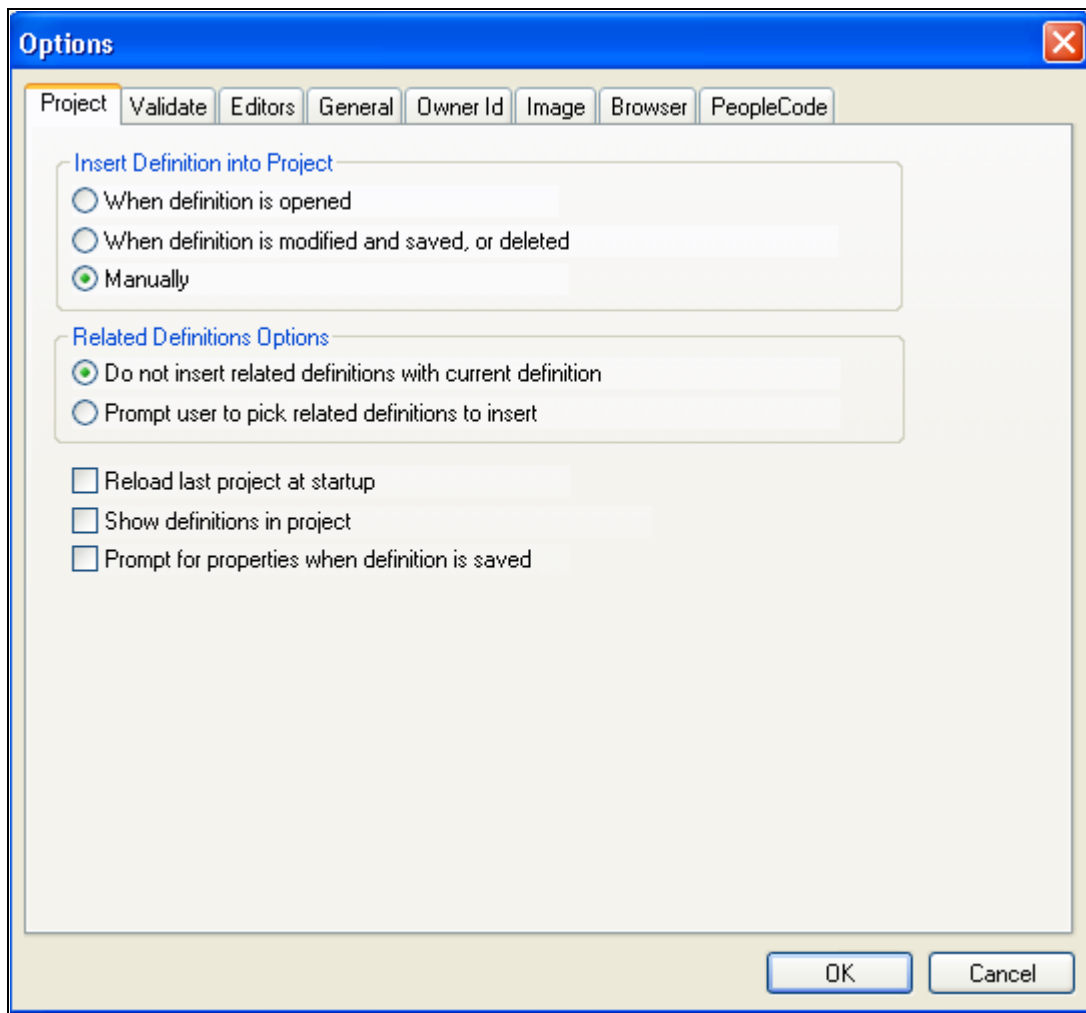
When you create a change project, you must set the appropriate update IDs and prerequisites.

See *PeopleTools 8.52: Change Assistant*, "Working with Change Packages," Creating a Change Project.

---

## Setting Project Options

Set processing options for all project operations in the Project Options dialog box:



Project options dialog box

To set project options:

1. Select Tools, Options.

The Options dialog box appears.

2. Select the Project tab in the Options dialog box.

Define when and how definitions are added to the project and how the project appears in the project workspace.

<b>When definition is opened</b>	Automatically inserts any definition that you open into the current project.
<b>When definition is modified and saved, or deleted</b>	Automatically inserts any definition that you save or delete into the current project.
<b>Manually</b>	Does not automatically insert any definition into the current project. This value is the default option. Use the Insert menu bar to insert a definition into the project.
<b>Do not insert related definitions with current definition</b>	Does not prompt you to include related definitions. Inserts only the specified definition, not related definitions.
<b>Prompt user to pick related definitions to insert</b>	After you select Insert, Current Definition into Project, a dialog box appears, prompting you to pick which related definitions to insert.
<b>Reload last project at startup</b>	When starting PeopleSoft Application Designer, the project that was last opened in the previous session opens automatically.
<b>Show definitions in project</b>	Displays a black mark next to icons for definitions that are explicitly in a project (Development view) to indicate that they are in the project. Attribute definitions, which are not explicitly part of the project, are not marked.
<b>Prompt for properties when definition is saved</b>	Prompts you with a definition properties dialog box before you can save the definition.

---

## Validating Projects

An important part of the PeopleSoft upgrade process involves validating your upgrade project. PeopleSoft Application Designer includes a validation utility to make sure that all definitions included in the project actually exist in your database.

To validate a project:

1. Close all active component definitions in the definition workspace.

Otherwise, the validation occurs in the component, rather than the project.

2. Select Tools, Options.

3. Select the Validate tab in the Options dialog box.
4. Select Validate project integrity.
5. Click OK.
6. Select Tools, Validate Project.

The validate utility runs a series of tests on the project or components and sends its results to the Validate tab in the output window. If errors are found, they are listed on this tab.

---

**Note.** The Validate WSRP compliance option applies to projects that contain PeopleSoft content intended for WSRP production.

---

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Using WSRP to Consume and Produce Remote Portlets," Validating WSRP Compliance.

---

## Setting System IDs

System ID functionality simplifies system ID maintenance for developers, especially for those working on mobile applications where the System ID field must be set to SYNCID on the record properties for each record of each component used in mobile applications.

You can set the system ID for an overall project or for the current record definition in the project workspace.

To set system IDs for a project:

1. Open the project.
2. Select Build, Set System IDs for Project

The Build Progress dialog box appears, displaying the progress of the command for the number of objects affected by this command.

### See Also

*PeopleTools 8.52: Global Technology*, "Selecting and Configuring Character Sets," Setting Data Field Length Checking



## Chapter 4

# Creating Field Definitions

This chapter provides an overview of field definitions and discusses how to:

- Create new field definitions.
- Use multiple labels.
- Open a field definition.
- View field properties.
- Change field definitions.
- Set custom field formats.
- Use the translate table.

---

## Understanding Field Definitions

Fields are the basic building blocks in your PeopleSoft system; in database terms, they represent columns in a table or view. Field definitions are stand-alone development definitions, defined in PeopleSoft Application Designer, and can be shared across multiple record definitions. Changes to field properties affect all records that include that field.

Each field definition includes attributes, such as data type, field name, long name, short name, field length, and various formatting values.

### ***Fields in Record Definitions***

By grouping fields that share a common theme, you build a record definition. For example, a department table (record definition) might include a department ID field, department name field, location field, and so on.

For each field definition that you add to a record definition, you define additional characteristics, called record field properties, that are specific to the way the field is used in that particular record. Record field properties are *not* shared with other records in which the field is used.

---

## Creating New Field Definitions

This section lists prerequisites and discusses how to:

- Create a new field definition.

- Specify attributes for a new field definition.
- Specify character field attributes.
- Specify long character field attributes.
- Specify number field attributes.
- Specify date field attributes.
- Specify time field attributes.
- Specify DateTime field attributes.
- Specify image field attributes.
- Specify ImageReference field attributes.
- Work with additional Oracle data type mappings.
- Work with additional Microsoft SQL Server 2005 data type mappings

## Prerequisite

You should be familiar with the general terminology and functions of PeopleSoft Application Designer.

## Creating a New Field Definition

To create a new field definition:

1. Click the New button on the PeopleSoft Application Designer toolbar.

The New dialog box appears.

2. Select Field.
3. Click OK.

A new character field opens in the definition workspace area.

4. Select one of the following field types from the Field Type drop-down list box.

---

**Note.** Make sure you name your fields according to accepted naming conventions.

---

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33.](#)

<b><i>Character (Char)</i></b>	Alphanumeric field of fixed length.
<b><i>Long Character (Long)</i></b>	Alphanumeric field of variable length used for textual entries, such as comments or descriptions.
<b><i>Number (Nbr)</i></b>	Positive numeric field of fixed length for which decimals are allowed.
<b><i>Signed Number (Sign)</i></b>	Positive or negative numeric field of fixed length for which decimals are allowed.
<b><i>Date (Date)</i></b>	Date field of constant length. The system edits dates and prohibits incorrect ones, such as day 42 or month 20. Date fields always store a four-digit year.
<b><i>Time (Time)</i></b>	Time field of a constant length. Built-in edits prohibit impossible times, such as hour 26, minute 70, or second 94.
<b><i>DateTime (DtTm)</i></b>	Date and time field of constant length. Built-in edits prohibit impossible dates and times, as defined in the individual Date and Time fields. DateTime fields always store a four-digit year.
<b><i>Image (Img)</i></b>	Image field to store images in a user-defined format, such as JPEG or GIF.
<b><i>ImageReference (Iref)</i></b>	Field in which to change an image dynamically at runtime using PeopleCode.
<b><i>Attachment (Att)</i></b>	Field that maps to a BLOB database type to hold the contents of a file attachment. However, you should use the existing field, FILE_DATA, if you need a BLOB column. For attachments, use the subrecord ATTACH_DETAIL, which contains FILE_DATA.

After you select a field type, a field definition window appears in the definition workspace. Use the field definition window to specify attributes to complete your new field definition.

## Specifying Attributes for a New Field Definition

Access the field definition window for a new field definition.

All field types share the following property settings.

---

**Note.** Multiple views of this dialog box are available, depending on which field type you select. Fields that are common to all views are documented first.

---

**Label ID**

Enter a unique identifier for the field that is less than 18 characters, without spaces, and in uppercase. Make this the default label by selecting the Def check box in the last column; otherwise, the first label is the default. You can set only one field label as the default.

---

**Note.** You can specify multiple labels for one field and change them at runtime using PeopleCode.

Label IDs must contain alphanumeric characters only and cannot consist entirely of spaces.

---

See [Chapter 4, "Creating Field Definitions," Using Multiple Labels, page 75.](#)

**Long Name**

Enter up to 30 characters. The name that you enter here appears as the RFT Long option for a page field. This label is also used for the tool tip, or hover text, for links at runtime if it is different from the assigned display value of the link.

---

**Note.** Do not use the \ (backslash) character in the long name of cubes and members. Unexpected behavior might occur.

---

**Short Name**

Enter up to 15 characters as an alternate name to use on pages and reports if you have insufficient space to display the long name. If you leave this field blank, the system automatically copies the first 15 characters of the long name into this field. The name that you enter here appears as the RFT Short option for a page field.

---

**Note.** If you name the field the same as the default label ID, you can identify it more easily when using PeopleCode.

---

---

**Note.** Do not use the \ (backslash) character in the short name of cubes and members. Unexpected behavior might occur.

---

**Def (default)**

When you enter the first label ID, the system automatically selects the Def check box. The system uses this default field if you have multiple labels. You can have only one default field.



**Not Used**

This check box is always read-only in the field definition properties dialog box. You can change its value using PeopleCode, causing the field to be:

- Skipped as part of the unique index.
- Exposed to PeopleCode for read and write operations.
- Ignored in Query.
- Ignored in nVision.
- Unavailable to users in search pages or list boxes if it is a search key or list box item.

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," SetDBFieldNotUsed.

**Chart Field**

This field is read-only in the field definition properties dialog box. It is used for character fields and by financial applications to specify a usage pattern. You can change the value using PeopleCode.

**Encrypt**

Select this option to encrypt the field using Oracle Transparent Data Encryption (TDE).

---

**Note.** The Encrypt option is disabled for field types that do not support encryption, for database platforms other than Oracle, as well as for Oracle database versions that do not support TDE.

---

See *PeopleTools 8.52: Data Management*, "Administering PeopleSoft Databases on Oracle," Encrypting Fields.

## Specifying Character Field Attributes

Use character fields for names, codes, and any fields with letter values. Also use them for numbers for which you want to contain formatting attributes and the data itself. Examples of such fields are those in which you enter telephone numbers and zip or postal codes. You cannot use character fields for calculations. You can enter decimal numbers in character fields, but you cannot use them mathematically until you convert them back to numeric fields.

You can specify character field attributes at design time, and you can access most of them at runtime with PeopleCode.

**Field Length**

Enter a whole number between 1 and 254 for the length of the field.

**Format Type**

Select one of the following values:

*Uppercase*: Converts the field value to uppercase and signifies that no other formatting options apply to this field. Use this option for code values, such as department ID, for which it does not matter whether the user enters the value in uppercase or lowercase. *Uppercase* is the default format.

*Mixedcase*: Stores uppercase and lowercase characters as entered. Use this option for fields that contain textual data, such as a department or company division name. For example, *Accounting - Receivables* is more readable than *ACCOUNTING - RECEIVABLES*.

*Numbers Only*: Forces entries to be numeric, which is useful for enforcing numeric values without redefining the field as a number field. This option automatically populates fields with leading zeros. For example, if the user enters *1* in a three-digit field, the system changes this value to *001*.

*SSN*: Formats the entry in U.S. Social Security Number format (999-99-9999). Define the field length as 9, even though the display length is 11. The system automatically adds dashes when formatting the field for display, but the number is stored in the database without the dashes.

*SIN*: Formats the entry in Canadian Social Insurance Number format (999-999-999) and performs the standard check-digit verification for SIN. Define the field length as 9, even though the display length is 11. The system automatically adds dashes when formatting the field for display, but the number is stored in the database without the dashes.

*Raw Binary*: Allows character fields containing embedded NULLs, such as encrypted values.

*Name*: Indicates that the field entry uses a PeopleSoft standard name convention: [lastname] [suffix],[prefix] [firstname] [middle name/initial]

The entry can contain alphanumeric characters, spaces, periods, hyphens, and apostrophes. Uppercase and lowercase characters are preserved as entered, that is, mixed case formatting is used automatically. Valid entries might include:

- O'Brien,Michael.
- Jones IV,James.
- Phillips MD,Deanna Lynn.
- Reynolds Jr.,Dr. John Q.
- Phipps-Scott,Ms. Adrienne.
- Knauft,Günter.

---

**Note.** For names that include Chinese, Japanese, or Korean (CJK) ideographic characters, you can use different standard name formats.

---

See *PeopleTools 8.52: Global Technology*, "Selecting and Configuring Character Sets," Using CJK Ideographic Characters in Name Character Fields.

*Postal Code North America:* Formats the entry to U.S. zip code format or Canadian postal code format. Canadian postal codes are edited to ensure that alphabetical and numeric characters are entered in the correct positions.

The following entry formats apply. Stored formats appear in parentheses following the entry format:

- 99999 (99999)
- 99999-9999 (99999-9999)
- 999999999 (99999-9999)
- A9A9A9 (A9A 9A9)
- A9A 9A9 (A9A 9A9)

*Postal Code International:* Allows the entry of international postal codes. You must enter at least 7 digits. If you enter 9 digits, the system assumes that the value is a U.S. zip code; if the entry is in A9A9A9 format, the system assumes that the value is a Canadian postal code. The system formats the value accordingly. For all other entries, no formatting is applied.

*Phone Num North America* (phone number North America): Formats the entry in North American telephone number format.

The following entry formats apply. Stored formats appear in parentheses following the entry format:

- 9999999 (999-9999)
- 999-9999 (999-9999)
- 9999999999 (999/999-9999)
- 999/999-9999 (999/999-9999)

*Phone Number International:* Allows the entry of international telephone numbers. If you enter 7 or 10 digits (and no other characters), the system assumes that the value is a North American phone number and formats it accordingly. For all other entries, no formatting is applied.

*Custom:* If you select this value, the Family Name and Display Name fields become available for entry. These selections enable you to apply additional formatting attributes that affect how the field appears on a page.

## Specifying Long Character Field Attributes

Depending on your database environment, the maximum length of a long character field ranges from several thousand characters to 32,767 characters (64,000 bytes). Using a long character field instead of a regular character field enables the user to insert a tab at runtime by pressing ctrl+tab.

The maximum Large Object (LOB) size allowed by PeopleTools is 50Mb. The following database platforms support a smaller maximum (in bytes):

- DB2/LUW (32,700)
- DB2/zOS (31,744)
- Sybase (32,000)

**Maximum Length**

To control the length of a long character field, enter the maximum number of bytes that you want the system to write to this field. If you do not specify a maximum length, or specify 0 (zero), you can enter an unlimited number of characters, depending on the capacity of your database system.

When working with fields, be aware that your chosen database system limits the amount of data contained in a field. For Oracle the field size is truncated to 1333 characters. Consider this limitation when choosing to use fields of a Long Character data type.

For Oracle, if the maximum length of a long field is less than 4001 bytes, the field can be stored more efficiently as a VARCHAR2(4000). To support the PeopleSoft Unicode implementation on the Oracle platform, we defined long columns with a specified length of 1330 or less as VARCHAR2 columns. If the length specified was greater than 1330, then the column was generated as a long VARCHAR or long raw, depending on the setting of the Raw Binary check box. This setting allows for the maximum possible usage of a tripled VARCHAR2 character length for Unicode-defined databases.

**Raw Binary**

Select to store data that is not converted when moving data between different systems. Raw binary is intended for binary data and byte strings. For example, raw binary can store graphics, sound, documents, and arrays of binary data.

## Specifying Number Field Attributes

When you create a new field, two field types are available for numbers:

- Number fields
- Signed number fields

Both field types are fixed in field length and allow the entry of positive numbers. Only signed numbers, however, allow the entry of negative numbers. Otherwise, both number field types are the same.

Unlike character fields, a number field can contain decimals but does not contain special formatting, such as the formatting for telephone numbers. Use number fields for calculations. If you use number fields for codes, you sacrifice flexibility to change your coding structure to alphanumeric in the future.

Field Type: 

Number

☐ Signed

Integer Positions: 

3

Decimal Positions: 

0

Field Labels

	Label ID	Long Name	Short Name	Def
1				<div></div>

Number Field type

**Integer Positions**

Specify the number of integer positions to the left of the decimal. If the number contains only decimal positions, leave this field blank.

**Decimal Positions**

Specify the number of decimal positions to the right of the decimal. If a number contains only integer positions, leave this field blank. In calculations, the system rounds up the result to the number of decimal positions defined here.

**Note.** Number fields cannot exceed 31 positions. Therefore, the sum of integer and decimal positions cannot exceed 31.

The field length notation differs between PeopleSoft Application Designer and SQL. If you specify a field length of 8 integer positions and 3 decimal positions in PeopleSoft Application Designer, SQL processes that as a length notation of 11.3.

SQL describes field length in terms of *precision* and *scale*. Precision is the total number of integer and decimal positions. Scale is the number of decimal positions. Thus, 11.3 means 8 integer positions and 3 decimal positions; 11.0 means 11 integer positions and 0 decimal positions.

When you enter 0 in the Decimal Positions field and enter 5 or 10 in the Integer Positions field, the RawBinary format option appears. The RawBinary format converts a number field value to a full-length 16- or 32-bit integer. See the example in the table that follows.

**Signed**

Select this check box for a field that is used for arithmetic calculations that might contain a negative value.

This table shows how the integer position affects the RawBinary storage format:

<i>Integer Position</i>	<i>Maximum Value</i>	<i>Number Stored As</i>
5	65535	16-bit integer
10	4294967295	32-bit integer

## Specifying Date Field Attributes

Date fields contain calendar dates. A date field has a field length of 10 and is maintained by the system. The default format of a date field is defined by the database and can be overridden by your browser settings.

### **Default to 2000 if year less than or equal to**

Specify the last two digits of a year, which is typically what users enter in a date field. If you enter *50*, the default century becomes 2000 when someone enters *0* through *50* as the year in a date field. If someone enters a number between *51* and *99* in a date field, the default century becomes 1900. The default value is *50*.

---

**Note.** When working with date and time fields on pages and search pages, carefully consider search criteria. Assume the date, time, and time zone fields are separate with the condition that the time is time zone-controlled by the time zone field and the time is associated with the date field. In this case, you need to provide date, time and time zone on the search page if any one of those search criteria is required.

---

## Specifying Time Field Attributes

Time fields are fixed-length fields (15 positions, format of HH:MI:SS.999999) that contain the time of day. The maximum time precision varies, depending on your database.

### **Time Formatting**

Specify the formatting for this field. HH represents hours, MI represents minutes, SS represents seconds, and 999999 represents microseconds.

---

**Note.** You can use %CLIENTDATE as the default constant for the Time field. %CLIENTDATE adjusts the date as appropriate to the time zone of the browser.

---

### **See Also**

*PeopleTools 8.52: PeopleCode Language Reference, "System Variables," %ClientDate*

*PeopleTools 8.52: PeopleCode Language Reference, "System Variables," %Date*

## Specifying DateTime Field Attributes

The DateTime field is a fixed-length field (26 positions, format of YYYY-MM-DD-HH-MI-SS.999999) that holds dates and times. The maximum date and time precision depends on your database.

---

**Note.** You can use %CLIENTDATE as the default constant for the DateTime field. %CLIENTDATE adjusts the date as appropriate to the time zone of the browser.

---

**See Also**

*PeopleTools 8.52: PeopleCode Language Reference, "System Variables," %ClientDate*

*PeopleTools 8.52: PeopleCode Language Reference, "System Variables," %Date*

**Specifying Image Field Attributes**

Use Image fields to store application data that takes the form of images. Image fields are functionally no different from any other type of field. Just as end-users employ character fields to store names and addresses, they can employ image fields to keep track of all types of digital pictures. This field enables you to store pictures of assets, company logos (in SQL tables), or scanned images of employees (as part of their personal data rows).

**Maximum Length**

Specify the maximum number of bytes available to store an image. If you do not specify a maximum length, the maximum length is determined by your database platform. If a user attempts to cut and paste an image that is larger than the maximum length or database capacity, an error message appears.

The following databases apply (maximum length appears in parentheses):

- DB2/zOS (up to 32 kilobytes, depending on the page size, the buffer pool, and the other data that is being stored in that row).
- Oracle (2 gigabytes).
- Sybase (2 gigabytes).
- SQL Server (2 gigabytes).
- Informix (2 gigabytes).
- DB2 UDB for Linux, UNIX, and Windows (32 kilobytes).

**Image Format**

Select an image format. Format determines the type of image that you can select and write to the database for this field. PeopleSoft Application Designer supports BMP, DIB, GIF, CUT, EPS, JPG, PCX, PCT, RLE, and TGA image types. At runtime, the user can upload BMP, DIB, GIF, and JPG images to the database.

**Specifying ImageReference Field Attributes**

Use the ImageReference field to store static images that you can change at runtime using PeopleCode. By associating image definitions with an ImageReference field on a page, you can display images dynamically. For example, to have different images appear on an employee profile depending on the status of the employee's current review, use the ImageReference field to reference the current review field and display the appropriate image.

## Working With Additional Oracle Data Type Mappings

If your site has upgraded to a PeopleSoft 9.x application and performed the appropriate conversion process during the application upgrade, different data type mappings exist between PeopleSoft field types and the underlying Oracle system.

See The upgrade documentation and release notes for your release for information about the conversion process.

The following table describes how the PeopleSoft field types map to the underlying Oracle data types:

<i><b>PeopleSoft Field Type</b></i>	<i><b>Preconversion Oracle Data Type Mapping</b></i>	<i><b>Post-Conversion Oracle Data Type Mapping</b></i>
Character	ANSI: VARCHAR2( <i>n</i> ) Unicode: VARCHAR2( <i>n</i> *3)	ANSI: VARCHAR2( <i>n</i> ) Unicode: VARCHAR2( <i>n</i> )  <b>Note.</b> For Unicode the VARCHAR2 column is interpreted as the number of Unicode characters due to init.ora parameter NLS_LENGTH_SEMANTICS being set to "CHAR".
Long Character	If $1 \leq n \leq 2000$ ANSI: VARCHAR2 Unicode: VARCHAR2( <i>n</i> *3) CHECK (Length <field name>) $\leq n$ If $n = 0$ or $n > 2000$ ANSI: LONG VARCHAR Unicode :LONG VARCHAR	ANSI: VARCHAR2( <i>n</i> ) when $1 \leq n \leq 4000$ CLOB when $n = 0$ , or $n > 1333$ Unicode: VARCHAR2 ( <i>n</i> ) when $1 \leq n \leq 1333$ CLOB when $n = 0$ , or $n > 1333$  <b>Note.</b> For Unicode the VARCHAR2 column is interpreted as the number of Unicode characters due to init.ora parameter NLS_LENGTH_SEMANTICS being set to "CHAR".
Long Character with Raw Binary attribute	ANSI: LONG RAW Unicode: LONG RAW	ANSI: BLOB Unicode: BLOB
Image	ANSI: LONG RAW Unicode: LONG RAW	ANSI: BLOB Unicode: BLOB
Attachment	ANSI: LONG RAW Unicode: LONG RAW	ANSI: BLOB Unicode: BLOB
Date	DATE	DATE



<b><i>PeopleSoft Field Type</i></b>	<b><i>Preconversion Oracle Data Type Mapping</i></b>	<b><i>Post-Conversion Oracle Data Type Mapping</i></b>
Datetime	DATE	TIMESTAMP
Time	DATE	TIMESTAMP

## Working With Additional Microsoft SQL Server 2005 Data Type Mappings

If your site has upgraded to a PeopleSoft 9.x application and performed the appropriate conversion process during the application upgrade, different data type mappings exist between PeopleSoft field types and the underlying Microsoft SQL Server 2005 system.

See The upgrade documentation and release notes for your release for information about the conversion process.

The following table describes how the PeopleSoft field types map to the underlying Microsoft SQL Server data types:

<b><i>PeopleSoft Field Type</i></b>	<b><i>Preconversion Microsoft SQL Server Data Type Mapping</i></b>	<b><i>Post-Conversion Microsoft SQL Server Data Type Mapping</i></b>
Character	ANSI: CHAR Unicode: NCHAR	ANSI: VARCHAR( <i>n</i> ) Unicode: NVARCHAR( <i>n</i> )
Long Character	ANSI: TEXT Unicode: NTEXT	ANSI: VARCHAR(MAX) Unicode: NVARCHAR(MAX)
Long Character with Raw Binary Image Attachment	IMAGE (ANSI and Unicode)	VARBINARY(MAX) (ANSI and Unicode)

---

## Using Multiple Labels

This section provides an overview of multiple labels and discusses how to:

- Add a new label.
- Delete a label.
- Select record field labels.

## Understanding Multiple Labels

You can define multiple label pairs for any field. By adding multiple labels to a field definition, you can expand its usability and create a more intuitive user experience at the same time.

For example, here you see the Date field with multiple labels.

The screenshot shows a window titled "DATE (Field)". At the top, "Field Type:" is set to "Date". Below this is a section titled "Field Labels" containing a table with the following data:

	Label ID	Long Name	Short Name	Def
1	BEGIN_DT	Begin Date	Begin	<input type="checkbox"/>
2	BIRTHDAY	Date of Birth	Birthday	<input type="checkbox"/>
3	DATE	Date	Date	<input checked="" type="checkbox"/>
4	FIRST_DAY_CLASS	First Day of Classes	Classes Begin	<input type="checkbox"/>
5	START_DT	Start Date	Starts	<input type="checkbox"/>
6				<input type="checkbox"/>

Below the table is a section titled "Default Century" with the text "Default to 2000 if year less than or equal" followed by a text box containing "50". At the bottom right, there are three checkboxes: "Not Used", "Encrypt", and "Chart Field", all of which are currently unchecked.

A date field showing multiple field labels

You could use this field in a student registration application and select the First Day of Classes label. You could also use this field in an application that tracks employee personal information, but select the Date of Birth label. In each case, you use the same field definition but you vary the label to reflect what the user sees and should understand about its context.

The underlying data structure for the field does not change when you change its label. At the database level, the column name in the table is DATE. You are merely manipulating the label to accommodate the context in which you use the field.

The system displays multiple labels in alphabetical order and sorted by label ID. You can sort the labels in a field by double-clicking the column header. The sorting is not saved, but it is helpful for quickly viewing the labels.

## Adding a New Label

Open the field definition in the definition workspace.

To add a new label, enter the label ID, long name, and short name in the last row. Each label can have related language labels. After defining the multiple labels, you can select appropriate labels when designing a record or page. You also can specify different labels in PeopleCode. To add a line feed for Long Name and Short Name, enter \n (backslash and the letter *n*). For example, *work \n month* adds a line feed between the words *work* and *month*.

When you create multiple labels, the first label ID is the default label. You can change the default label by selecting the Def check box for a different row. Only one label can be the default.

---

**Note.** Each label ID must be unique so that each field definition label can be identified separately, and it must be entered using uppercase letters with no spaces.

---

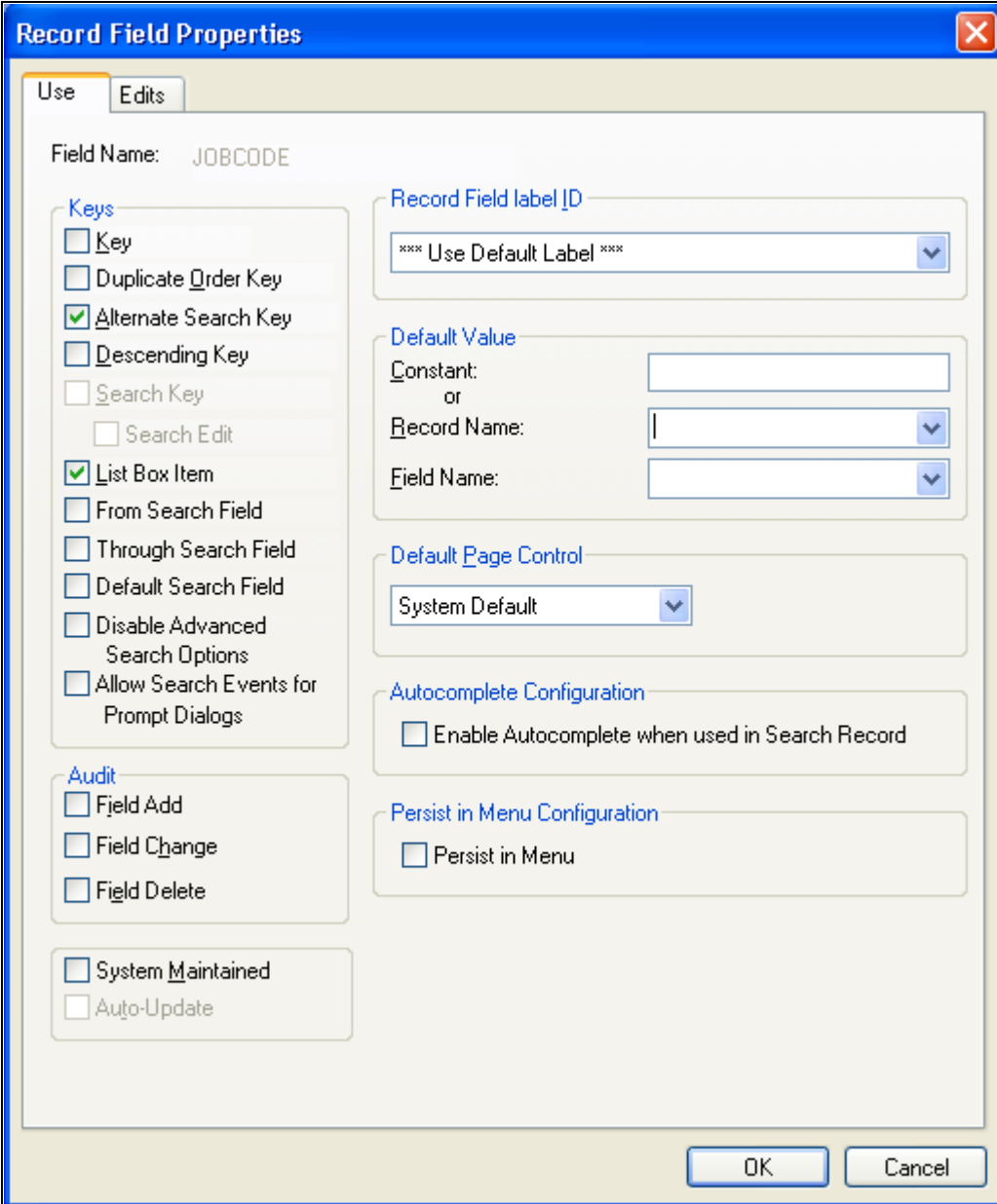
## Deleting a Label

Open the field definition in the definition workspace.

To delete a label, right-click a field label row and select Delete Label from the pop-up menu. You cannot delete labels that are currently in use. To determine where the label is used, select the label and select Find Definition Reference from the pop-up menu. The results appear in the Find Definition Reference output window.

## Selecting Record Field Labels

Access the Record Field Properties dialog box.



The image shows the 'Record Field Properties' dialog box with the 'Use' tab selected. The 'Field Name' is 'JOBCODE'. The 'Keys' section has 'Alternate Search Key' checked. The 'Record Field label ID' is set to '\*\*\* Use Default Label \*\*\*'. The 'Default Value' section has 'Constant' checked. The 'Default Page Control' is 'System Default'. The 'Autocomplete Configuration' has 'Enable Autocomplete when used in Search Record' unchecked. The 'Persist in Menu Configuration' has 'Persist in Menu' unchecked. The 'Audit' section has 'Field Add', 'Field Change', and 'Field Delete' unchecked. The 'System Maintained' and 'Auto-Update' options are also unchecked. The 'OK' and 'Cancel' buttons are at the bottom right.

**Record Field Properties**

Use | Edits

Field Name: JOBCODE

**Keys**

- ☐ Key
- ☐ Duplicate Order Key
- ☒ Alternate Search Key
- ☐ Descending Key
- ☐ Search Key
  - ☐ Search Edit
- ☒ List Box Item
- ☐ From Search Field
- ☐ Through Search Field
- ☐ Default Search Field
- ☐ Disable Advanced Search Options
- ☐ Allow Search Events for Prompt Dialogs

**Record Field label ID**

\*\*\* Use Default Label \*\*\*

**Default Value**

Constant:

or

Record Name:

Field Name:

**Default Page Control**

System Default

**Autocomplete Configuration**

☐ Enable Autocomplete when used in Search Record

**Persist in Menu Configuration**

☐ Persist in Menu

**Audit**

- ☐ Field Add
- ☐ Field Change
- ☐ Field Delete

☐ System Maintained

☐ Auto-Update

OK Cancel

Record Field Properties dialog box

For the record definition, the Record Field label ID drop-down list box is available on the Use tab of the Record Field Properties dialog box. The drop-down list box displays the label ID, short name, and long name that you set in the field definition. The default selection is *\*\*\* Use Default Label \*\*\**. The record field label changes automatically whenever you change the default label in the field definition.

## Opening an Existing Field Definition

You can open a field definition in several ways:

- Select File, Open from the menu.

From this dialog box, you can view all available fields in the database. You can open all fields of a certain type or from a specific project, or you can open a specific field or set of fields.

- Clone an existing definition.

To create a field definition with attributes similar to an existing one:

1. Open an existing field definition.
2. Select File, Save As.
3. Enter a new name and click the OK button.

- Open a field from the project workspace.

Double-click a field from the project workspace to open a field definition window in the definition workspace.

- Open a field from the record.

Open an existing record definition. When the record definition window opens, you can:

1. Select a field.
2. Right-click the field.
3. Select View Definition from the pop-up menu.

---

## Viewing Field Properties

You can view field properties in several ways:

- While in the field definition workspace, click the Properties button.
- Select File, Definition Properties.
- While in a record definition:
  1. Select the field.
  2. Right-click the field name.
  3. Select View Field Properties from the pop-up menu.

The General tab in the Field Properties dialog box provides a place for you to enter comments documenting the purpose of the field definition. The Owner ID drop-down list box enables you to select a valid owner, as in application project or role. The Last Updated group box contains information—date, time, and user ID—about the last time that the field definition was modified.

The Translate Values tab is available only for character fields.

**See Also**

Chapter 4, "Creating Field Definitions," Using the Translate Table, page 92

---

## Changing Field Definitions

This section provides an overview of the effects of changing field definitions and discusses how to:

- Change field types.
- Rename field definitions.
- Delete field definitions.
- Print field definitions.

## Understanding the Effects of Changing Field Definitions

When you develop new record definitions or modify existing ones, you might need to change characteristics or attributes for a field. If you change attributes in a field definition, the change affects every occurrence of the field in every record definition. So, before you change basic attributes in a field definition, consider how it affects all of the record definitions in which the field occurs. If the change is not appropriate for every occurrence of the field, consider defining a new field instead.

If a change is still needed, you should determine the extent of the change by finding out how many records are affected. Do this by using the Find Definition References tool. Open the field definition and select Edit, Find Definition References or right-click and select the option from the pop-up menu. If you are in a record definition window, another method is to select the field, right-click the field name, and select Find Definition References--Field from the pop-up menu.

The Find Definition References tab in the Output window displays a list of all projects, record definitions, page definitions, messages, channels, file layouts, and PeopleCode programs in which the field is used. Double-click a line of output to open the corresponding definition in the definition workspace.

---

**Note.** Changing the name or length of a field definition requires that you modify the underlying SQL table. Do this either by running the build process and specifying Create Tables or Alter Tables, or by carrying out an action issued by your system administrator. For example, if you change the length of a field definition and 30 record definitions contain the same field name, then you must create or alter 30 records.

---

## Changing Field Types

Change a field type by opening a field and selecting another field type from the drop-down list box. However, this feature is intended as a design-time change—that is, before your tables are created in the database. While you can convert some field types, the results vary depending on your database platform.

---

**Note.** If you change a field from one type to another, you must rebuild all records containing that field by running the Alter process. Use Find Definition References to find all instances of the changed field.

---

See [Chapter 7, "Administering Data," Altering Tables, page 181.](#)

## Renaming Field Definitions

You can use either the Rename Definition dialog box or the Open Definition dialog box to rename a field definition.

### ***Using the Rename Definition Dialog Box***

To rename a field definition from the Rename Definition dialog box:

1. Select File, Rename to open the Rename Definition dialog box.
2. Select Field as the definition type.
3. Enter the name of the field.

If you are not sure of the name or the correct spelling, use the search criteria fields to locate the field to rename. The list displays field names that match your search criteria.

4. To rename a field in the list, select the field and click Rename or double-click the field.

The field name becomes available for entry, and you can enter a new name.

5. When you finish typing the new name, press enter or click the Rename button.

The Rename Field dialog box appears. Select the records in which you want the field to be renamed. The De-select All button toggles to Select All.

6. Click OK.

### ***Using the Open Definition Dialog Box***

To rename a field definition from the Open Definition dialog box:

1. Select File, Open to access the Open Definition dialog box.

This dialog box enables you to set up search criteria for locating field names.

2. Select Field as the definition type.

If you have additional information, use the Name field. You can further narrow your search criteria by selecting a field type or the name of the project that contains the field to delete.

3. Click the Open button or press enter when you have completed your search criteria.
4. You can rename a field in the Open Definition dialog box by selecting the field, right-clicking the field name, and selecting *Rename*.

You can also select the field name and click it a second time. The field name becomes available for entry and you can enter a new name.

5. When you finish typing the new name, press enter or click the Rename button.

## ***Results of Renaming Field Definitions***

When you rename a field, the system automatically changes all occurrences of it, including data in tables with columns named FIELDNAME where the data matches the field being renamed.

The only references that are *not* renamed are the text portions of SQL functions, such as SQLExec and Scroll Select. If you already used SQL Create to create the underlying tables for the record definitions that contain the field that you renamed, you must recreate or alter those tables. If the tables contain data that you want to preserve, use the SQL Alter function to rename the database table fields.

## **Deleting Field Definitions**

You can use either the Delete Definition dialog box or the Open Definition dialog box to delete a field definition.

### ***Using the Delete Definition Dialog Box***

To delete field definitions by using the Delete Definition dialog box:

1. Select File, Delete to open the Delete Definition dialog box.  
Set up search criteria for locating field names.
2. Select *Field* as the definition type.
3. If you have additional information, use the Name field.
4. Select a field type or the name of the project that contains the field that you want to delete.

This selection further narrows your search criteria.

5. When your search results appear in the output window, delete the field definition by selecting the field and clicking the Delete button or double-clicking the field name.

---

**Note.** You cannot delete a field that is currently used in any records. Before you delete a field definition, you must first remove it from records in which it appears.

---

### ***Using the Open Definition Dialog Box***

To delete field definitions by using the Open Definition dialog box:

1. Select File, Open to access the Open Definition dialog box.  
This dialog box enables you to set up search criteria for locating field names.
2. Select *Field* as the definition type.
3. If you have additional information, enter it in the Name field.
4. You can further narrow your search criteria by selecting a field type or the name of the project that contains the field that you want to delete.
5. When you have completed your search criteria, click Open or press enter.



6. Right-click the field that you want to delete from the list of field names displayed.
7. Select Delete from the pop-up menu.

## Printing Field Definitions

If you plan to make changes to your field definitions, you should view your field definition before you proceed. You can view it on your screen or print it to a one-page report that combines information from several resources. This method can also serve as a paper audit trail to document your database.

To print a field definition, select File, Print from the field definition window (or click the Print button in the toolbar) to open the Print dialog box, where you specify printing options. You can also print field definitions from the Open Definition dialog box. On the list resulting from your search, right-click the name of the field definition that you want to print, and select Print from the pop-up menu to view your printing options.

---

## Setting Custom Field Formats

This section provides overviews of custom field formats and format notation and discusses how to:

- Change format families.
- Edit formats.
- Use family options.
- Test formats.

## Understanding Custom Field Formats

PeopleTools has formatting provisions at both the field and page levels. The field format specifications affect internal values that are stored in the database, and the page formats affect the visual presentation of values.

Several formatting options are available for U.S. Social Security Numbers, Canadian Social Insurance Numbers, postal codes, telephone numbers, and various currencies, dates, and times. Custom field formats extend these formatting provisions to support the same types of data in other countries. They also support other types of formatted information, such as bank codes, credit card numbers, part numbers, or serial numbers.

## Understanding Format Notation

Stored and display formats are defined by strings that contain *lexical*, *literal*, and *meta* characters:

- Lexical characters delimit character expressions and designate interpretation rules.
- Literal characters represent only themselves.
- Meta characters represent a class of characters.

Use the format symbols in the following tables to create the format notation for your own custom field formats:

<b>Meta Symbol</b>	<b>Description</b>
#	<p>Digit placeholder.</p> <p>If the number has more digits to the right of the decimal point than there are # symbols to the right in the format, the system truncates the number to as many decimal places as there are # symbols to the right.</p> <p>If the number has more digits to the left of the decimal point than there are # symbols to the left in the format, the system displays the extra digits.</p> <p>If the number has fewer digits to the right of the decimal point than there are # symbols to the right of the decimal point in the format, the system adds spaces.</p> <p>If the number has fewer digits to the left of the decimal point than # symbols to the left of the decimal point in the format, the system also adds spaces.</p> <p>Example format: ###.##</p> <p>Input: 1234.567 - Matches? Yes - Output: 1234.56</p>
0 (zero)	<p>Digit placeholder.</p> <p>Follows the same rules as for #, except that if the number has fewer digits than there are 0s in the format, the system displays the extra 0s.</p> <p>Example format: 000.00</p> <p>Input: 1.2345 - Matches? Yes - Output: 001.23</p>
(period)	<p>Decimal point.</p> <p>This symbol determines how many digits (0 or #) appear to the right and left of the decimal point.</p> <p>If the format contains only # symbols to the left of this symbol, numbers less than one appear with a decimal point.</p> <p>This symbol has meaning only in conjunction with the # and 0 symbols.</p> <p>Example format: ###.##</p> <p>Input: .12345 - Matches? Yes - Output: .12</p>
9	<p>Required numeric placeholder.</p> <p>If the number does not have the same number of digits as there are 9s, the system displays an error message.</p> <p>Example format: 999</p> <p>Input: 123 - Matches? Yes - Output: 123</p> <p>Input: 12 - Matches? No</p>

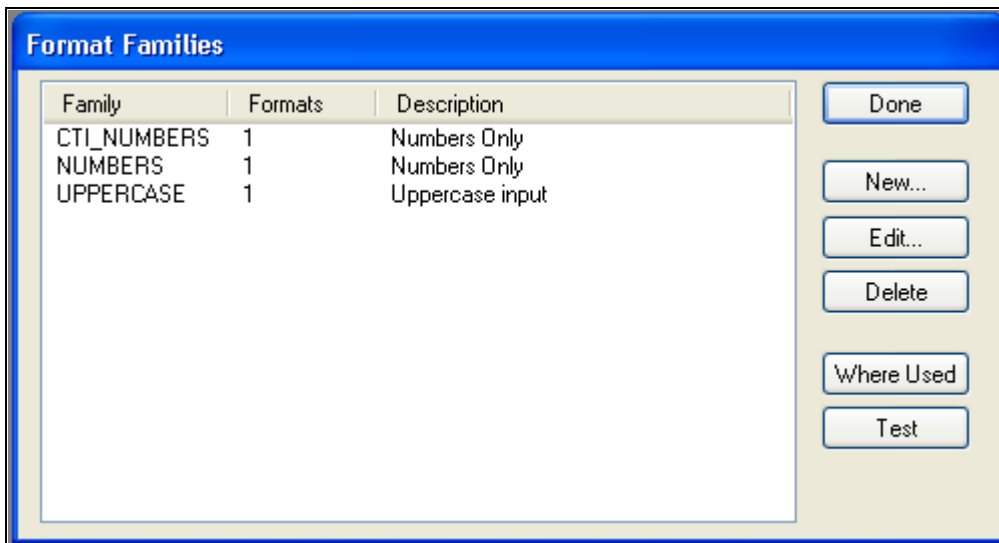
<b>Meta Symbol</b>	<b>Description</b>
<b>A</b>	Matches any alphabetic character: <i>A–Z</i> or <i>a–z</i> . Example format: <i>AA</i> Input: <i>Sd</i> - Matches? Yes - Output: <i>Sd</i> Input: <i>4A</i> - Matches? No
<b>Z</b>	Matches any alphabetic or numeric value: <i>A–Z</i> , <i>a–z</i> , or <i>0–9</i> . Example format: <i>ZZ</i> Input: <i>3g</i> - Matches? Yes - Output: <i>3g</i> Input: <i>A3C</i> - Matches? No
<b>@</b>	Matches any character. Example format: <i>@@@</i> Input: <i>Iq?</i> - Matches? Yes - Output: <i>Iq?</i>

<b>Lexical Symbol</b>	<b>Description</b>
<b>*</b>	Matches zero or more occurrences of this character expression. Example format: <i>a*b</i> Input: <i>b</i> - Matches? Yes - Output: <i>b</i> Input: <i>ab</i> - Matches? Yes - Output: <i>ab</i> Input: <i>aab</i> - Matches? Yes - Output: <i>aab</i>
<b>+</b>	Matches occurrences of this character expression. Example format: <i>a+b</i> Input: <i>b</i> - Matches? No Input: <i>ab</i> - Matches? Yes - Output: <i>ab</i> Input: <i>aab</i> - Matches? Yes - Output: <i>aab</i>
<b>[ ]</b>	Denotes a character expression that matches the input character if the input character is the same as any character in the list enclosed by square brackets. If the expression matches, the input character appears. The system interprets all characters enclosed in square brackets as literal characters. Example format: <i>[ab]c</i> Input: <i>ac</i> - Matches? Yes - Output: <i>ac</i> Input: <i>bc</i> - Matches? Yes - Output: <i>bc</i>

<b><i>Lexical Symbol</i></b>	<b><i>Description</i></b>
{ }	<p>Denotes a character expression that matches the input character if the input character matches any character in the list enclosed by curly braces.</p> <p>If the expression matches, the first character in the list (not the input character) is copied to output. All characters enclosed in curly braces are interpreted as literal characters.</p> <p>Example format: {ab}c</p> <p>Input: <i>ac</i> - Matches? Yes - Output: <i>ac</i></p> <p>Input: <i>bc</i> - Matches? Yes - Output: <i>ac</i></p>
?	<p>This expression is optional. It is copied to output only if it appeared in input.</p> <p>Example format: a?b</p> <p>Input: <i>ab</i> - Matches? Yes - Output: <i>ab</i></p> <p>Input: <i>b</i> - Matches? Yes - Output: <i>b</i></p>
!	<p>This expression is optional. It is copied to output regardless of whether or not it is matched.</p> <p>Example format: a!b</p> <p>Input: <i>ab</i> - Matches? Yes - Output: <i>ab</i></p> <p>Input: <i>b</i> - Matches? Yes - Output: <i>ab</i></p>
~	<p>This expression is optional. It is copied to output regardless of whether or not it is matched.</p> <p>Example format: a~b</p> <p>Input: <i>ab</i> - Matches? Yes - Output: <i>b</i></p> <p>Input: <i>b</i> - Matches? Yes - Output: <i>b</i></p>
\	<p>Example format: a\b</p> <p>Input: <i>ab</i> - Matches? No</p> <p>Input: <i>a?b</i> - Matches? Yes - Output: <i>a?b</i></p>
( )	<p>Groups expressions.</p> <p>Example format: (abc)!99</p> <p>Input: <i>abc12</i> - Matches? Yes - Output: <i>abc12</i></p> <p>Input: <i>12</i> - Matches? Yes - Output: <i>abc12</i></p>

## Changing Format Families

Custom field formats enable you to create your own format definitions using format notation and apply them to fields. These formats are organized into format families, which can include one or more unique formats.



Format Families dialog box

To change format families:

1. Select Tools, Miscellaneous Definitions, Field Formats to access the Format Families dialog box.

The Format Families dialog box lists the defined format families and enables you to edit, delete, or create new families. You can also determine where each family is used in your applications and perform tests.

2. To define a new format family, click the New button.

The New Family dialog box appears.

3. Enter a name for your new format family, and click OK.

The Formats dialog box appears.

4. In the Stored group box, enter a name and format.

This information defines how the format family is to be stored in the database and printed in reports.

5. Enter the format notation that you want to use for storing the data in the Format field.

Because a stored format cannot exist alone, a display format is always required. If you are adding to a family that has existing stored formats, a display name and format pair is optional. PeopleSoft Application Designer automatically places all of the current display formats in the new stored format. The display format default is the stored format that you can redefine later.

6. In the Display group box, enter a name for the format.

This information defines how the format family appears online.

7. Enter the format notation that you want to use for displaying the data in the Format field.

8. Click the Add button to add the new format family, and then click OK when you are finished.

## Max Display Length

Max Display Length (maximum display length) enables you to override the default length that PeopleSoft Application Designer uses to determine the length of the edit field in custom format. This setting is useful when the displayed data is longer than the defined field length. For example, you might store data as 999 (field length in the database is 3 characters), but the display format of the data is 9-9-9 (5 characters).

PeopleSoft Application Designer uses the field length of 3, by default, for the edit field, which is not long enough to hold the five characters of the reformatted data. By entering 5 in the Max Display Len (maximum display length) field, the page uses five characters for the edit field without requiring that the field length in the database be five characters long.

## Editing Formats

The Edit Formats dialog box presents a tree where you define the stored and display formats for a family and all options and attributes. This dialog box enables you to:

- Add, update, and delete stored and display formats.
- Enter a description for the family.
- Specify options, such as uppercase and smart punctuation, for the family.

Click OK to save changes.

Formats dialog box

This section discusses how to:

- Add a new stored format.

- Add a new display format to a stored format.
- Update a display format.
- Delete stored formats.

### ***Adding a New Stored Format***

To add a new stored format:

1. Open the format family to which you want to add a format.
2. Select the format family name.
3. Enter the stored and display names and format notation.
4. Click Add.

The new format appears in the tree below the existing formats.

### ***Adding a New Display Format to a Stored Format***

To add a new display format to a stored format:

1. In the tree, select the stored format name to which you want to add a display format.
2. In the Display group box, enter a new name and format notation.
3. Click the Add button to add the display format.

The new display format appears in the tree, attached to the stored format to which it belongs.

### ***Updating a Display Format***

To update a display format:

1. Select the display format in the tree view.

---

**Note.** Clicking the Update button applies only to display formats.

---

2. Select a display format and enter the enabled fields.
3. Click Update to update the display.

### ***Deleting Stored Formats***

To delete a stored format:

1. Select the display format.
2. Click the Delete button.

PeopleSoft Application Designer displays a warning message indicating that deleting a display format might strand stored data. Select Yes to continue with the deletion.

---

**Note.** Stored formats cannot be deleted directly. Therefore, all display formats for a specific stored format must be deleted first to ensure that data is not stranded in the database under an invalid format. When the last display format is deleted for a stored format, the stored format is automatically deleted.

---

## Using Family Options

Family options include:

- Smart punctuation
- Make uppercase

### **Smart Punctuation**

Smart punctuation automatically adds or removes punctuation characters to the data that users enter. For instance, if you enter a phone number, *8005551212* and press the tab key to exit the entry field, the system reformats the value as *800-555-1212*.

The system uses the punctuation characters specified in the punctuation list to strip out unwanted input and to modify the pattern specified before applying the pattern to the data.

---

**Note.** Smart punctuation and the standard set of characters are enabled by default.

---

You can modify the standard set of characters by selecting *Custom* in the options drop-down list box and typing another set of punctuation characters to use. Smart punctuation processes the characters defined in the standard set or custom set as follows:

- The input has all characters in the punctuation set stripped out—similar to ignoring white spaces.
- The pattern has all characters in the punctuation set expanded if they are not found in the input.

To do this, enclose each punctuation character in the punctuation set with "[ *puncchar* ]!".

- The formatter runs on the modified data to apply the modified pattern.

---

**Warning!** Use meta characters and lexical characters in the punctuation sets with caution. Step 2 of the transformation process modifies the format pattern so that PeopleSoft Application Designer might sometimes convert special characters to literals (when they are inside the left or right brace).

---

For example:

- Input: *(800)555-1212*.
- Display pattern: *999-999-9999*.
- Punctuation set: *()-*.
- Stored pattern: *9999999999*.
- The *"(" , ")"*, and *"-"* are stripped out of the input because they appear in the punctuation set.

Now the input looks as follows: *8005551212*.



The display pattern is modified to `999[-]/!999[-]/!9999` because `"-"` is in the punctuation set. The transformed data (`8005551212`) is matched against the display pattern of `999[-]/!999[-]/!9999`. The match is successful, so the raw data is transformed into `800-555-1212`.

Now, the input data has successfully matched the expected display format. The next step is to transform this data into what we want stored. In this case, the stored format has no dashes, as in `9999999999`.

The transformed data (`800-555-1212`) is stripped of the smart punctuation characters (`8005551212`), and the pattern of the stored format is `9999999999`. A match is made so that the system stores `8005551212` in the database.

Finally, the stored data is reformatted back to the display using `8005551212` and the display pattern. Output to the display is reformatted to `800-555-1212`.

### ***Make Uppercase***

Custom format fields support mixed case. With uppercase activated, PeopleSoft Application Designer transforms the data into uppercase when you press the tab key to exit the field.

## **Testing Formats**

The Test Format dialog box enables you to test a specific format or unformatted path for a family without having to build a page:

Test Format dialog box

### **Stored Format Name and Display Format Name**

Select a stored and display format pair to test.

### **DB->Display**

Click this button to process the input field like data from the database. The system applies the stored format pattern to the input, and then applies the display format pattern to the result. The output appears in the Transform field, which is the same output displayed for a field using this format.

**Display->DB**

Click this button to process the input field like user input from a page. The system applies the display format pattern to the input, and then applies the stored format pattern to the result. The transformed output appears in the Transform field. This output reflects what is stored in the database if the operation is successful. On an actual page, when the user presses the tab key to exit the field, the system calls the DB->Display function to reformat the user input for the display.

---

## Using the Translate Table

This section provides an overview of the Translate table and discusses how to:

- Add values to the Translate table.
- Change translate values.
- Delete translate values.
- Save the Translate table.

## Understanding the Translate Table

The Translate table is a prompt table, similar to an all-purpose data dictionary, that stores values for fields that do not need individual prompt tables. As a general rule, store field values in the Translate table if the field meets the following criteria:

- Field type is *Character*.
- Field length is 1 to 4 characters.
- Field values consist of a relatively small, static set of values that are not maintained by the user.
- No other fields relate to this field.

---

**Note.** If the only values for a field are *Y* (yes) and *N* (no), you do not need to enter them in the Translate table. The Translate table comes with a field for *Y* and *N* named PSYESNO. When you select Yes/No Table Edit, the system automatically points to the PSYESNO field in the Translate table.

---

### ***Example of When to Use the Translate Table***

Consider the relative attributes of a department ID field and a gender field in a company database. In the following table, Department ID is not a good case for using the Translate table because it requires its own prompt table. However, the Gender field is a good case for using the Translate table:

<b><i>Characteristic</i></b>	<b><i>Department ID</i></b>	<b><i>Gender</i></b>
Field type	Character.	Character.

<b>Characteristic</b>	<b>Department ID</b>	<b>Gender</b>
Field length	Longer than 3 characters.	1 character.
Field values	Could be many values, 20 or more.	Two possible values.
Other related fields	Related to other fields, like department manager.	Not related to other fields.
Requires maintenance	Yes, users might have to add departments.	No, gender is a static value.

### **Translate Table Attributes**

The system maintains the structure of the Translate table and you supply the data. This table contains the following fields:

<b>Field Name</b>	<b>Description</b>
FIELDNAME	Field name, such as ABSENCE_TYPE.
LANGUAGE_CD	Language code.
FIELDVALUE	Value for the field.
EFFDT	Effective date.
VERSION	Internal version number (system-maintained).
EFF_STATUS	Status: active or inactive.
XLATLONGNAME	Thirty-character description; used as a label on pages and reports.
XLATSHORTNAME	Ten-character description; used as a label on pages and reports.
LASTUPDDTTM	DateTime field showing the last time that a field was updated.
LASTUPDOPRID	User ID of the user who most recently updated the field.

If you must define more information about a field than the preceding attributes, create a separate prompt table for the field instead of adding it to the Translate table.

When the user presses the prompt button to prompt for valid values in a field on a page, the system displays a list of the translate values for a field.

Each value in the Translate table has an effective date, and the date must be earlier than the effective dates of any rows that reference the value. The translate values delivered by PeopleSoft all have an effective date of January 1, 1900. Choosing a different value for translate values that you add enables you to quickly differentiate delivered values from your values.

**Field Properties**

General Translate Values

Field Name: QE\_ABSENCE\_TYPE

	Value	Active	Eff Dt	Long Name	Short Name
1	CNF	<input checked="" type="checkbox"/>	01/01/1900	Conference Attendance	Conference
2	DIS	<input checked="" type="checkbox"/>	01/01/1900	Disability	Disability
3	DSC	<input checked="" type="checkbox"/>	01/01/1900	Disciplinary	Disciplinary
4	JUR	<input checked="" type="checkbox"/>	01/01/1900	Jury Duty	Jury Duty
5	MAT	<input checked="" type="checkbox"/>	01/01/1900	Maternity/Paternity	Maternity
6	OTH	<input checked="" type="checkbox"/>	01/01/1900	Other	Other
7	PER	<input checked="" type="checkbox"/>	01/01/1900	Personal	Personal
8	REL	<input checked="" type="checkbox"/>	01/01/1900	Religious	Religious
9	SCK	<input checked="" type="checkbox"/>	01/01/1900	Sick	Sick
10	VAC	<input checked="" type="checkbox"/>	01/01/1900	Vacation	Vacation

Add  
Change  
Delete

Last Updated  
Date/Time: 05/21/1997 4:35:10PM  
By User: PPLSOFT

OK Cancel

Field Properties dialog box: Translate Values tab

## Adding Values to the Translate Table

Access the Add Translate Table Value dialog box.

To add translate values to the Translate table:

1. From a field definition window, select File, Definition Properties.

You can also right-click and select *Field Properties*, or press alt+enter.

2. From a record field definition window, right-click the field and select View Translates from the pop-up menu.
3. Select the Translate Values tab.

The Translate Values dialog box displays existing values for the field and enables you to add, change, or delete values. In the Last Updated box, you see information—date, time, and user ID—about the last update for the selected translate value.

4. Click Add to define a new value.
5. Enter the field value, effective date, long name, and short name that your users enter on the page.

#### **Field Value**

Enter the translate value for the field. The system automatically sorts values in ascending order as you enter them. If you enter three translate values with field values *A*, *T*, and *C*, they appear in the Translate Values dialog box as *A*, *C*, *T*.

#### **Effective Date**

Enter the date on which you want this value to take effect. The default value is today's date. If you want the effective date to predate all rows on your database, enter *01011900* (January 1, 1900).

---

**Note.** The Add button only allows you to add a new, distinct translate value. To add a different effective date for the same translate value, select the translate value and use the Change button. As long as the effective date is changed, the system saves the modified entry.

---

#### **Inactive**

When a value for a field becomes obsolete, you should select the Inactive check box rather than deleting the value. Deactivating a value enables fields in the database that still contain the value to use the correct long and short names. If you delete an obsolete code from the Translate table and you still have records in the database that contain that value, then you must change all of those values to active values.

#### **Long Name**

Enter up to 30 characters.

#### **Short Name**

Enter up to 10 characters. If you leave this field blank, the system automatically copies the first 10 characters of the long name into this field. Use mixed case to improve readability.

---

**Note.** The recommended character length of translate short name values is six. You can configure PeopleSoft Application Designer to alert you if the recommended length is exceeded.

---

See [Chapter 2, "Using PeopleSoft Application Designer," Setting PeopleTools Options, page 22.](#)

## Changing Translate Values

Sometimes the meaning of a translate value changes, but you still need to retain both values in the Translate table. For example, suppose that your PROFICIENCY field has a value of *E* (extremely high). You have been in production for five years when upper management decides to change the meaning of the value to *Exceptional* as of January 1, 1998.

You can accommodate this type of change by creating a second active entry for the same translate value. The first value is valid in the time range between the effective dates of January 1, 1990, and December 31, 1997. The second entry is valid from January 1, 1998, onward. You do not want to delete the existing entry because the database might contain rows that predate January 1, 1997, and contain this value. You might also maintain the existing definition for historical reporting.

To change an existing translate value, select the value and click Change or double-click the value. The system displays the Change Translate Table dialog box, in which you can make the necessary changes.

## Deleting Translate Values

Use caution when deleting a translate value, unless you are removing an invalid value that was entered by mistake. To deactivate a value because it is no longer used or its meaning has changed, either change the long and short names, or change the status to *Inactive*. To delete the value, select it and click the Delete button. The system removes the row.

## Saving the Translate Table

After you have made all of your changes to the Translate table, you must save the *entire field definition*. The system has no save option specifically for translate values. Click OK from the Field Properties dialog box, and then save the field definition by clicking the Save button or selecting File, Save.

To save the Translate table, you must be authorized to modify field definitions. A special Translates Only security access level is available for fields; it allows authorized users to update translate values but not other field attributes.

## Chapter 5

# Planning Records, Control Tables, and TableSets

Planning is the first step in the application development process. As a system designer, you must consider how to store, retrieve, manipulate, and process data that is stored in the tables in your application database.

This chapter provides an overview of the planning process and discusses how to share tables.

---

## Understanding the Planning Process

This section discusses:

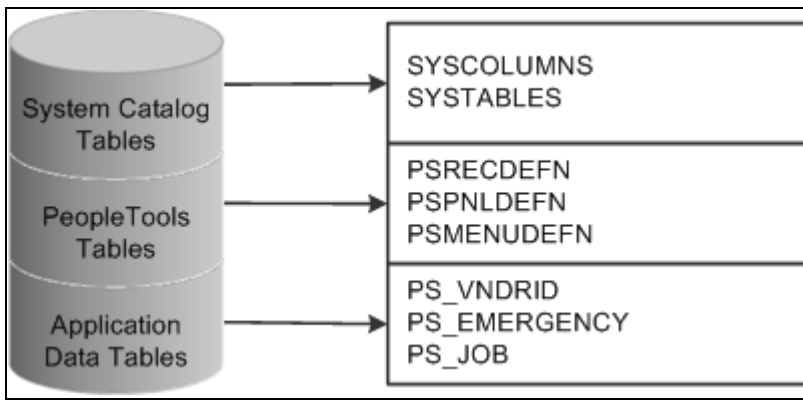
- Table-based systems.
- Normalized relational databases.
- Record definition planning.
- Effective dates.
- Control tables.
- TableSets.

## Table-Based Systems

PeopleTools-based applications are table-based systems. A database for a PeopleTools application contains three major sets of tables:

- System Catalog tables store physical attributes of tables and views, which your database management system uses to optimize performance.
- PeopleTools tables contain information that you define using PeopleTools.
- Application Data tables house the actual data that your users enter and access through PeopleSoft application pages.

This diagram shows PeopleSoft database tables and sample names:



Tables in a PeopleSoft database

Like a spreadsheet, each of these tables contains columns and rows. Columns define the structure of how the data is stored. Rows represent the actual data that is stored in the database.

Every time that you create a new definition using PeopleTools, the system inserts rows of data into various PeopleTools tables. The entries in these tables determine the online processing of the system and what happens during imports. PeopleSoft maintains the structure of these tables. You maintain data in the PeopleTools tables related to definitions that you create or adapt using PeopleTools. You can view the PeopleTools tables in the PPLTOOLS project using the PeopleSoft Application Designer.

To create the application data tables that store the rows of data that your *users* manipulate:

1. Create a record definition.

In doing so you determine the structure of the table, the characteristics of the fields, and any online processing that you want to occur behind the scenes when a user enters data.

2. Apply the SQL Create option to build the SQL table in which your application data will reside based on a subset of parameters in your record definition.

During this process, the system automatically gives the application data table the same name as your record definition, prefaced with *PS\_*.

## Normalized Relational Databases

To better understand the structure of your PeopleSoft system, you should be familiar with the concept of a normalized relational database. A *normalized* table adheres to standards that are designed to improve the productivity of the database user. Normalization makes the database more flexible, allowing data to be combined in many different ways.

The standards for a normalized database are called *forms*, such as *first normal form*, *second normal form*, and so on.



### ***First Normal Form***

The first normal form requires that a table contain no repeating groups of nonkey fields. In other words, when you set up a record definition, if you encounter a field that could have multiple occurrences, put that field in a separate record that is subordinate to the primary record definition (a child record). This setup allows unlimited occurrences of a repeating field rather than a specified number. Each row of data is uniquely identified by a primary key, which can be a single field or a group of fields that, when concatenated together, form a unique key.

For example, look at the record definition structure of the tables that we use to schedule exam times for different locations in our training database. Here are the necessary fields, in order of importance:

- LOCATION
- EXAM\_DT
- EXAM\_TIME

You know that you have multiple exam dates and times per location. You could set up record definitions to accommodate this data as follows:

<b><i>Record Definition</i></b>	<b><i>Fields</i></b>	<b><i>Key?</i></b>
LOCATION	LOCATION	Yes
EXAM	LOCATION EXAM_DT	Yes Yes
EXAM_TIME	LOCATION EXAM_DT EXAM_TIME	Yes Yes Yes

Because multiple exam dates per location might exist, we added exam dates to the second record definition (child record) that is subordinate to the first (parent) record. Similarly, because one date could have multiple exam times, exam times are located in a third record definition that is subordinate to the second.

### ***Second Normal Form***

The second normal form dictates that every nonkey field in a table must be completely dependent on the primary key. If two fields make up the key to a table, every nonkey field must be dependent on both keys together. For example, if a table has the Employee ID and the Department ID fields as keys, you would not put the Department Name field in the table because this field is dependent only on the Department ID field and not on the Employee ID field.

### **Third Normal Form**

The third normal form is a corollary to the second; it requires that a nonkey field not be dependent on another nonkey field. For example, if a table is keyed by the Employee ID field and Department ID is a nonkey field in the table, you would not put the Department Name field in the record because this field is dependent on a nonkey field (Department ID). You would find the Department Name field only in the table that is keyed by the Department ID field, not in any other table that contains Department ID.

With the third normal form, you store shared fields in tables of their own and reference them elsewhere. For example, you would not put the Department Name field in every record definition in which the Department ID field appears. Instead, you would create a prompt table of department IDs and department names. Similarly, you would create a prompt table of job codes and job titles instead of putting a job title in every employee's record.

---

**Note.** When designing record definitions, you should adhere to the third normal form to increase flexibility and reduce data redundancy.

---

## **Record Definition Planning**

Before you begin to create record definitions, you should know how you plan to use the record definition, the fields that it will contain, special edits that you would like to see performed on the record definition, or specific fields in the definition.

You define two layers of information:

- Record level
- Field level

At the record level, determine the ultimate purpose of the record definition and how it will be used in the system. Is it destined to define an underlying SQL table to hold data? Are you building a view to join or retrieve information from other tables? Do you need a temporary work record where you can store derived data?

You can audit record-level changes, as opposed to individual fields contained in the record definition, which is an efficient alternative if you plan to audit several fields. You can also establish a more sophisticated use of record definitions, such as sharing information in TableSets and multilanguage controls, at the record level.

At the field level, plan the details of what types of fields to add. Should they be character fields or number fields? Should automatic formatting be used? What are the keys to the data stored in the database? Which fields should you audit? Do you want to specify prompt tables so that users can select from lists of valid values that are stored elsewhere in the database?

In most cases, if you are creating a record definition for a SQL table, you do not have to worry about record-level definitions for parameters and conditions. Unless you change how a record definition is used, the system automatically assumes that you are defining a record definition for an underlying SQL table.

## **Effective Dates**

Effective dates enable you to keep historical, current, and future information in tables. You can use the information to review the past and plan for the future. Three types of effective dates are available:

<b>Future</b>	Data rows that have effective dates that are after the system date, which is usually today's date.
<b>Current</b>	Data row with the most recent effective date that is closest to today's (system) date, but not a future date. Only one row is the current row.
<b>History</b>	Data rows that have effective dates before the current data row.

The EFFDT (Effective Date) field has special properties related to the processing of effective dates on rows and should be used only when needed.

Unlike regular date fields, which you can use anywhere in the system, use the EFFDT field only in record definitions for which you want to maintain data history—future, current, and past—to store rows of data in sequence. Using EFFDT fields only for this circumstance enables you to store multiple occurrences of data based on when it goes into effect.

For effective-dated rows, you can have multiple occurrences of future and history but only one current row of data.

EFFDT is almost always a key and almost never a list item. Activate the Descending Key attribute so that the row with the most recent effective date appears first on pages. You might enter %DATE (current system date) as the default constant for this field.

---

**Note.** Alternatively, you can use %CLIENTDATE as the default constant for the date field. %CLIENTDATE adjusts the date as appropriate to the time zone of the browser.

---

To enable you to track an accurate history of your effective-dated information, the system invokes special logic when you access a record definition that contains EFFDT. The action that you select dictates whether you can access the row type and what you can do with each type of row:

<i><b>Action Type</b></i>	<i><b>View</b></i>	<i><b>Change</b></i>	<i><b>Insert New Rows</b></i>
Update/Display	Current, Future	Future Only	Effective Date Greater Than the Current Row
Update/Display All	History, Current, Future	Future Only	Effective Date Greater Than the Current Row
Correction	History, Current, Future	All Existing Rows	Add New Rows with No Effective Date Restrictions

---

**Note.** For records that do not contain EFFDT, all actions (Update/Display, Update/Display All, and Correction) operate the same way: they retrieve all existing rows for the specified keys.

---

When you run a page with effective-dated records and you insert a row, the system copies the contents of the previous row into the new row to save you keying time. In a large effective-dated table, you do not want to reenter all of the data when only a single field changes. Also, anytime you insert an effective-dated row using PeopleCode, the system copies the contents of the previous row.

**Effective Status**

In prompt tables, EFF\_STATUS (Effective Status) usually accompanies EFFDT. When used with EFFDT, it is part of the mechanism that enables the system to select the appropriate effective-dated rows.

You can also use EFF\_STATUS by itself as a simple status field, but do not change the translate values. They must be *A* (active) and *I* (inactive) for EFFDT to work properly. If you need a status field with different values, use or define a different field.

**Effective Sequence**

The EFFSEQ (Effective Sequence) field serves different purposes, depending on whether it is paired with EFFDT. If EFFSEQ is not paired with EFFDT, then EFFSEQ has no special function and can be used as a simple sequencing field wherever you need one.

If EFFSEQ is paired with EFFDT, it enables you to enter more than one row with the same effective date. You assign a unique sequence number to each row that has the same effective date. Do not make EFFSEQ a required field; a value of *unrequired* allows the first EFFSEQ to be zero. Select Display Zero in the page definition to have zeros appear on the page.

For example, suppose that you want to enter both a transfer and a pay rate change for an employee, and both actions are effective on the same day. Enter the transfer on the job data pages as usual, and leave the Effective Sequence Number field as *0* (zero). Then, insert a row to enter the change in pay rate. This time, the effective date is identical to the previous row, but enter *1* in the Effective Sequence Number field.

**See Also**

*PeopleTools 8.52: PeopleCode Language Reference*, "System Variables," %ClientDate

*PeopleTools 8.52: PeopleCode Language Reference*, "System Variables," %Date

**Control Tables**

Control tables store information that controls the processing of an application. This type of processing might be consistent throughout an organization (in which case the entire organization shares the same control information), or it might be used only by portions of the organization for more limited sharing of data.

**Sharing One Set of Common Values**

The first type of sharing is to create one table that everyone shares; it stores common information that is valid for *all* users, such as a country table to store country codes or a department table to store department codes. Such control tables are ordinarily maintained centrally because the data is shared throughout the entire organization.

Sharing Common Values in Overlapping Plans

What do you do if the codes that are stored in a table are valid only for *some* users? Consider benefit plans, for example. Typically, you store information for benefits plans in a plan table. However, not all plans are valid for all employees; their validity might depend on whether the employee is full-time or part-time, union or nonunion. Some plans might overlap; some might be appropriate for all employees and others only for some. In a relational database, you do not want to define the same plan value—and associated data—more than once.

In this case, you can easily resolve the problem by using two tables. The first is the plan table, which stores the relevant data for each plan. The second table defines which plans are valid for various benefit programs or groups of plans. For example, one benefit program might be valid for nonunion employees, and another benefit program might contain the plans as negotiated with a union.

This table shows how you might set up the two tables to reflect the benefits offered:

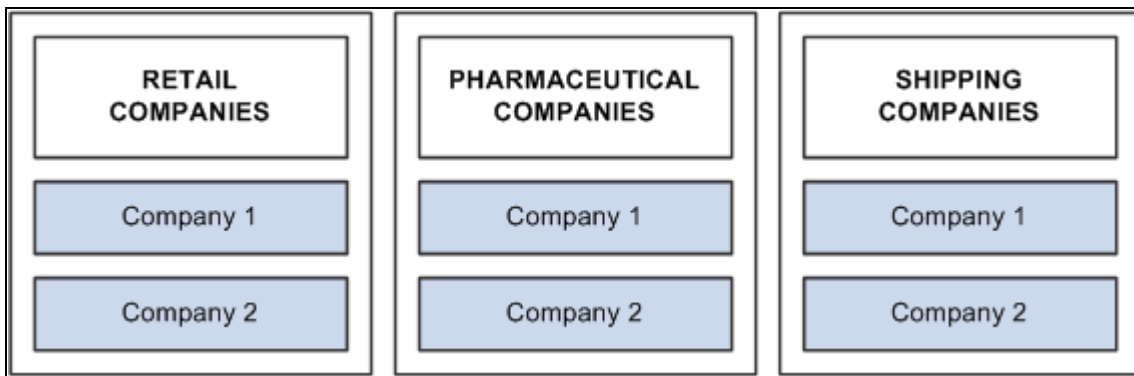
Benefit Program Table		Benefit Plan Table			
Key	Valid Values	Key	Description	Field	Field
Nonunion Program	Plan 1	Plan 1	Health	...	...
	Plan 2	Plan 2	Life	...	...
Union Program	Plan 2	Plan 3	Savings	...	...
	Plan 3	Plan 4	Health-Union	...	...
	Plan 4			...	...

These tables are ordinarily centrally maintained because the data is shared by various groups in the organization.

TableSets

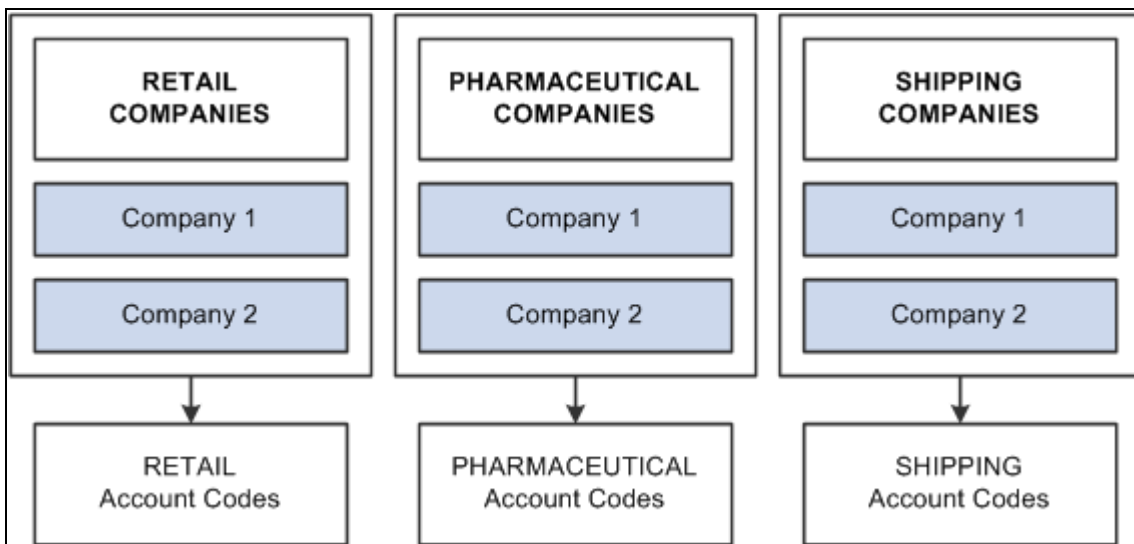
When *none* of the information stored in control tables is valid for all users, but the structure of these common tables is the same, you can set up a way to share multiple sets of values. For example, a multicompany organization must store completely different sets of accounting codes for its various operating entities, and the data for these accounting codes is maintained in a set of relevant control tables. The actual data values differ, but the structure of the control tables remains the same. PeopleTools enables you to share sets of values in a control table through *TableSets*.

To better understand TableSets, consider an organization that has two retail stores with common accounting codes, two pharmaceutical firms with another set of accounting codes, and two shipping firms with yet another set of codes.



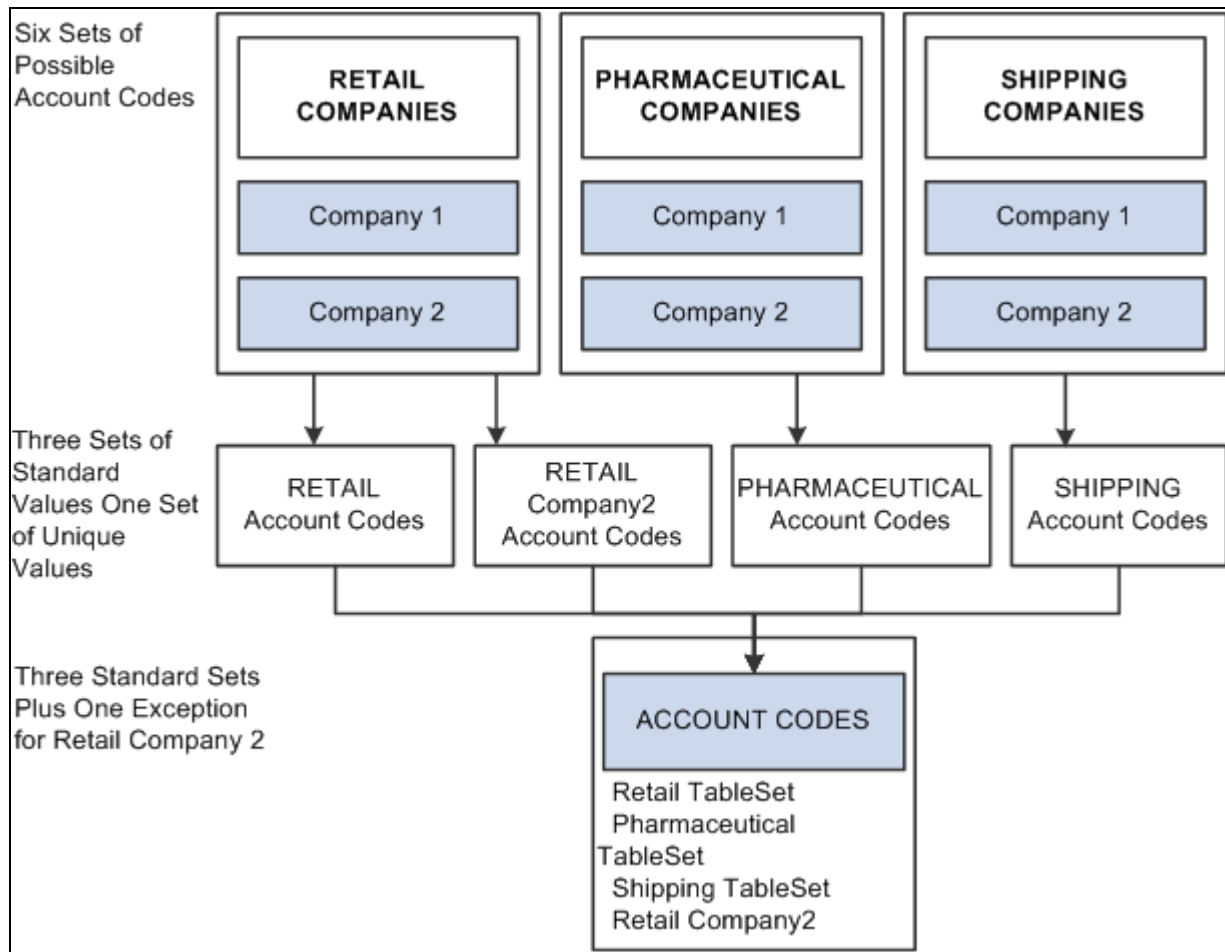
Maintaining multiple account codes for multiple companies

If each of these companies has completely different accounting codes, you can establish six different sets of account codes to be maintained by each company. If they all have exactly the same accounting codes, you can limit them to one set of values. However, the reality is usually somewhere in between. That is, there is one set of account codes for each type of business: retail, pharmaceutical, and shipping. Rather than having six different companies maintaining separate copies of this common data, you can reduce the number to three sets:



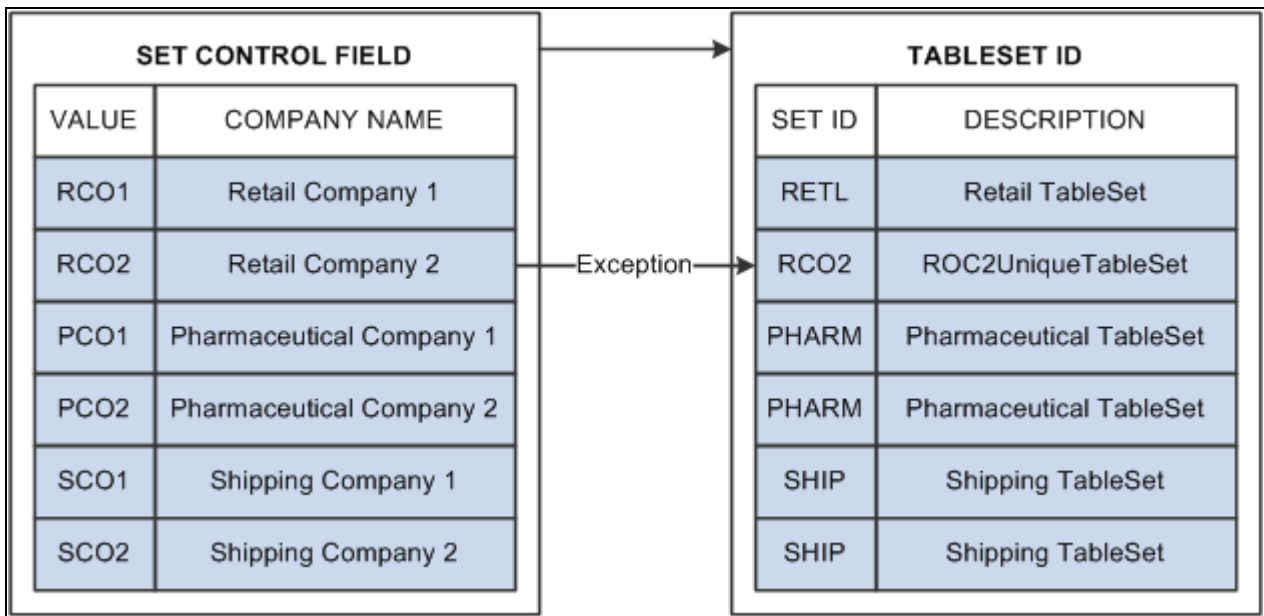
Sharing multiple account codes among companies

You can also handle exceptions. Suppose that Retail Company 2, a recently acquired company, has its own unique set of account codes. A separate set of values should be maintained for this company as an exception to the retail rule:



### Sharing account codes among companies with exceptions

When you share tables in PeopleTools applications, you add the `setID` field as an additional key or unique identifier to the table that you want to share. This key identifies the sets of information in the table that are shared by multiple companies or business units under your corporate umbrella. You then specify a *set control* field, which identifies which fields map between the original key and the TableSets. You can specify any field that logically identifies the TableSet. In this example, you might assign the `Company` field as the set control:



Linking set controls and TableSets

### Sharing Groups of Record Definitions

While this example illustrates how you might share data values for a single table—Account Codes—you typically share data that is stored in many tables that are based on the same TableSets. To minimize the overhead of defining TableSets, you can define *record groups* that share table data in a similar manner. For example, rather than using the TableSets that you establish for accounting codes solely for the Accounting Code table, you can group all accounting related tables into one record group.

### TableSets and PeopleSoft Applications

Some PeopleSoft applications already take full advantage of TableSets and table sharing. Throughout the PeopleSoft Financials and HCM product lines, TableSets are used extensively, in most cases triggered by business unit.

## Sharing Tables

This section provides an overview of table sharing and discusses how to:

- Add the SetID field to record definitions.
- Define set control fields as the fields controlling the assignment of TableSets.
- Modify set control fields.
- Create setIDs.
- Define record groups to identify the tables and subordinate (child) tables that are affected.
- Define TableSet controls.



- Share trees.

## Understanding Table Sharing

To share tables, you must modify the record definitions for both the table that you want to share and the one that contains the set control field. Using the online PeopleTools Utilities for TableSets, you define the terms or controls for sharing.

For example, the PeopleTools database holds data for three companies: one U.S.-based parent company (PST) and two sister companies (CCB in the U.S. and VNB in Canada). In this organization, all of the U.S.-based companies share one set of accounting codes, and all of the Canadian-based VNB companies share another:

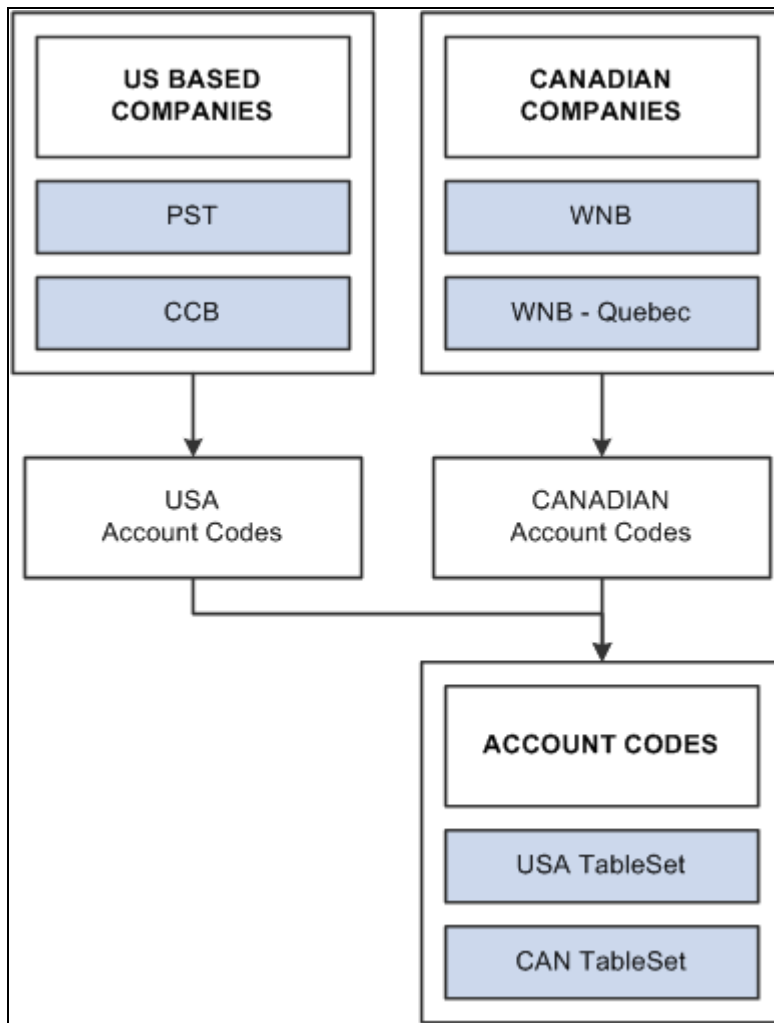
<i><b>U.S. Account Codes</b></i>	<i><b>Canadian Account Codes</b></i>	<i><b>Description</b></i>
123456789	123456789	Teller
987654321	987600000	Customer Service
CCB-4476-EXTSAL-USA	VNB-4476-EXTSAL-CDN	Extraordinary Salary Employees
CCB-4476-REGSAL-USA	VNB-4476-REGSAL-CDN	Regular Salary Employees

These two groups of companies must maintain parallel sets of accounting codes and, in some cases, use the same code, uniquely identified for U.S. or Canadian companies.

You must set up the Account Code table so that users who access that page or prompt for valid values see only the values for their respective companies, depending on whether they are located in the U.S. or Canada.

To do this:

1. Modify the Account Code table, which will be shared, and the Company table, the values of which (CCB, PST, VNB) control the TableSets that are used.
2. Create two TableSet IDs:
  - USA: For U.S. companies.
  - CAN: For Canadian companies.



TableSets and set control fields

### Assigning Set Control Fields

If you plan to use the Table Sharing feature to add an additional high-level key to identify common sets of values and handle exception values, you enter a set control field. The set control field determines which set of values appears, based on how you define table sharing.

Table sharing is usually applied in multicompany environments, in which you might want to share certain tables in some operating entities while allowing others to retain control over the contents of their own. Use the feature to maintain a single table in which you can store multiple sets of values to be used by different entities in your organization.

When you create a record definition that will be shared, you add the field `setID` as a high-level key. The `setID` value is controlled by the key that you select in the Set Control Field drop-down list box in the Record Properties dialog box. These `setIDs` identify groups of tables to be shared and enable each business entity to identify whether it wants to use the corporatwide table or maintain its own. Enter the name of the key as your set control field for each record definition that will be shared.

For example, if you have several companies in your organization, many of which share the same set of account codes, add a SetID field to the Account table record definition. In this definition, enter *Company* as the set control field. The set control enables different companies to access the account values stored in the same Account table. As each company defines its set controls, it has the option to use the shared Account table or maintain control of its own.

## Adding the SetID Field to Record Definitions

Add the setID field to the record definition for table sharing purposes.



Num	Field Name	Type	Len	Format	Short Name	Long Name
1	<b>SETID</b>	Char	5	Upper	SetID	SetID
2	TREE_NODE	Char	20	Upper	Node	Tree Node
3	EFFDT	Date	10		Eff Date	Effective Date
4	<b>SEQNUM</b>	Nbr	3		SeqNum	Sequence number
5	EFF_STATUS	Char	1	Upper	Status	Status as of Effective D
6	EMAILID	Char	70	Mixed	Email ID	Email ID
7	<b>DISTID</b>	Char	30	Mixed	ID	Distribution ID
8	<b>DISTIDTYPE</b>	Char	3	Upper	ID Type	Distribution ID Type

Adding setIDs to record definitions

To add a setID:

1. Open the record definition for the table that you want to share.
2. Select New, Field, Character.
3. Enter the special field that PeopleTools provides for table sharing:

**SETID**

SetID is a 5-character field. Add it to the top of your record definition list, and define it as a required key, search and list item, with a prompt table edit against the PeopleTools SETID\_TBL.

## Defining Set Control Fields

After you add the setID field to your record, identify the set control field in Record Properties.

To define the set control field:

1. Select Edit, Definition Properties from the menu.
2. Select the Use tab.

3. Select the field that identifies the appropriate setID in the Set Control Field drop-down box.

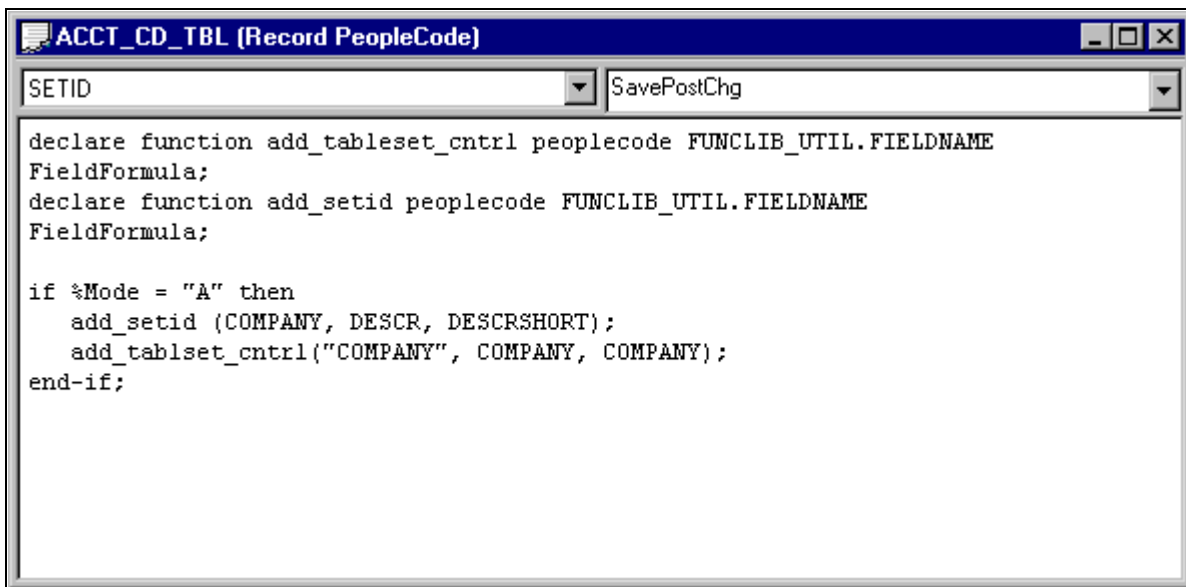
See [Chapter 5, "Planning Records, Control Tables, and TableSets," Understanding Table Sharing, page 107.](#)

4. Click OK to save your changes.
5. Use the SQL Alter process to alter your underlying SQL table.
6. When the system prompts you for a default setID, enter the most common setID for your company.

## Modifying Set Control Fields

If you are sharing tables or using a set control field that is not already defined as such in your PeopleSoft application, you must make some modifications to the record definition for the table in which you store values for your specified set control field. For example, all PST company codes are stored in the COMPANY\_TBL.

Here, you must add PeopleCode to the set control field so that each time you add a new value, the system populates the set controls in PeopleTools utilities, where you assign the appropriate TableSets.



### Attaching PeopleCode to your set control field

This program assigns a default setID with the same value as the Set Control Field value, for each new COMPANY row that you add to the COMPANY\_TBL. For example, when you add a row for CCB in the Company table, it creates a default setID of CCB. This way, each value that you add is assigned a unique TableSet until you assign shared TableSets in the Utilities window.

You can easily clone and modify this program to change the ADD\_SETID and ADD\_TABLESET\_CNTRL statements to reflect your table sharing objectives. Alternatively, if you are working with existing TableSet functionality, you can point to a model setID that is delivered with your PeopleSoft application.

**See Also**

Chapter 7, "Administering Data," Altering Tables, page 181

**Creating SetIDs**

After you prepare for sharing your table, you define the setIDs that logically group information. When you define setIDs, you create groups that might naturally share more than one table. For example, although CCB and VNB are planning to share only the ACCT\_CD\_TBL now, the setIDs that you set up might define logical divisions in the organization that serve as the basis for sharing all accounting-related tables. Describe your sets to give them as broad an application as is practical in your organization.

If possible, use descriptions that denote the shared table and set control field. For example, for CCB and VNB, we added two TableSet IDs, identifying them as related to accounting codes for companies:

<i>setID</i>	<i>Description</i>	<i>Short Description</i>
USA	Accounting Codes - USA Co.s	AcctCd USA
CAN	Accounting Codes - CAN Co.s	AcctCd CAN

You create SetIDs on the TableSetID page in PeopleTools, Utilities, Administration, TableSet IDs.

**See Also**

*PeopleTools 8.52: System and Server Administration, "Using PeopleTools Utilities," TableSet IDs*

**Defining Record Groups**

Record groups are a set of logically and functionally related control tables and views. They help enable TableSet sharing, which eliminates redundant data entry and ensure that it is applied consistently across all related tables and views.

In the record group table, you group the record definitions for the tables that you want to share and any dependent record definitions. If you add a table to a PeopleSoft application, an appropriate record group might already be defined. However, if you add new business functions, you might need to add a new record group for the tables that you define. You access the record groups table in PeopleSoft Pure Internet Architecture on the Record Group page.

**See Also**

*PeopleTools 8.52: System and Server Administration, "Using PeopleTools Utilities," Record Group*

## Defining TableSet Controls

As you add values for your set control, the system automatically populates the TableSet Controls 1 table with default values. You define TableSet controls in PeopleTools, Utilities, Administration, TableSet Control using the browser. Here you can enter the set control value for which you want to assign a TableSet.

For example, we use Company as our set control field, and PST, CCB, and VNB are values that we want to add to the Company table. The system automatically assumes that each new set control value maintains its own set of tables and does not enable table sharing. So, the default values are the same as the set control value. In this case, when we added the value CCB to the Company table, the system populated the TableSet controls with a default setID of CCB.

Because CCB is a U.S.-based company that should share the same U.S. accounting codes, we must change the defaults to USA.

### See Also

*PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," TableSet Control

## Sharing Trees

To share trees as well as tables and views, complete TableSet Controls 2 in the same way that you did TableSet Controls 1. For example, if PST and CCB want to share one department tree for organizational security and VNB another, you assign the appropriate setID for each value.

### Defining TableSet Controls for Trees

To define TableSet controls for trees:

1. Select PeopleTools, Utilities, Administration, TableSet Control in PeopleSoft Pure Internet Architecture.
2. Enter the name of the set control value for which you want to assign a TableSet on the search page.

For example, we use Company as our set control field, and PST, CCB, and VNB are values that we want to add to the Company table.

Because you already assigned a default setID in TableSet Controls 1, the system displays the default setID that you assigned to this field value. If you create another TableSet for sharing trees, you can change this value.

3. Specify the tree name.

Click the prompt button next to the Tree Name field for a list of only the tree definitions with the same set control field (which has PostSave PeopleCode that recognizes it as a TableSet control field).

This prompt list comes from a SQL view of tree definitions with those set control fields that have not already been associated with a tree group.

4. In each tree row, specify the appropriate setID.
5. Save your changes.

### ***Viewing All Record Definitions Associated With a Set Control***

You can view all record definitions that are associated with a set control on the Record Group tab of the TableSet Controls page.

### ***Example of Set Controls***

After you set up and define all of your set controls, you can see the results by looking at pages on which you reference codes that are stored in a shared table. For example, on Administer Workforce, Use, Job Data 1 and 2 pages, Account Code is an optional field that is edited against the ACCT\_CODE\_TBL. In this component, the system identifies the company based on the department to which an employee is assigned. For example, Simon Schumacher is in department *10100*, which is associated with the company *CCB*.

Select the Job Data 2 tab and enter *8001* or *Simon Schumacher* in the search record dialog box to retrieve a CCB row, for which the company is defined in the department table. Here, you can see that Simon works in department *00001*, which identifies him as a CCB employee. If you click the prompt button next to the Account Code field, the system displays only those rows in the Account Code Table that are associated with USA, the setID for CCB.

Alternatively, if you access the job row for Joan Avery, an employee of VNB, and click the prompt button for the same Job Data 2 Account Code field, the system retrieves only values that are associated with CAN, the setID for VNB.





## Chapter 6

# Creating Record Definitions

This chapter presents an overview of record definitions and discusses how to:

- View record definitions.
- Save record definitions.
- Name record definitions.
- Create new record definitions.
- Open existing record definitions.
- Manipulate fields in record definitions.
- Set record properties.
- Set record field properties.

---

## Understanding Record Definitions

Fields that are grouped together as a unit are *record definitions*. A record definition represents what the underlying SQL database tables look like and how they process data. Two tabs exist in record definitions.

- Use the Record Fields tab to access the fields that compose the record definition.

QE\_DEPT\_TBL (Record)

Record Fields

Record Type

	Num	Field Name	Type	Len	Format	Short Name	Long Name
	1	SETID	Char	5	Upper	SetID	SetID
	2	DEPTID	Char	10	Upper	DeptID	Department
	3	EFFDT	Date	10		Eff Date	Effective Date
	4	EFF_STATUS	Char	1	Upper	Status	Status as of Effective D
	5	DESCR	Char	30	Mixed	Descr	Description
	6	DESCRSHORT	Char	10	Mixed	Short Desc	Short Description

Record Fields tab

- Use the Record Type tab to specify way that the record definition is to be used, for example a SQL table or view, or perhaps a temporary table.

The screenshot shows a window titled "QE\_DEPT\_TBL (Record)". It has two tabs: "Record Fields" and "Record Type", with "Record Type" being the active tab. Inside the "Record Type" tab, there is a group box labeled "Record Type" containing seven radio button options: "SQL Table", "SQL View", "Dynamic View", "Derived/Work", "SubRecord", "Query View", and "Temporary Table". The "SQL View" option is selected. To the right of these options, there is a section for "Non-Standard SQL" with a "Table Name:" label and an empty text input field. Below that is a "Build Sequence No:" label with a text input field containing the number "1". At the bottom of this section is a button labeled "Click to open SQL Editor".

Record Type tab

## Understanding Record Types

Select the Record Type tab to view the record definition types. Select from these types:

### SQL Table

Select to define a record definition that has a corresponding physical SQL table in the database. Create this table when you run the Build Operation from the Build menu. This value is the default setting.

**SQL View**

Select to define a record definition that corresponds to a SQL view, which is not a physical SQL table in the database but fields from one or more SQL tables that are reorganized into a different sequence. SQL view provides an alternate view of information that is stored in tables.

To create the SQL view, click the Click to open SQL Editor button, enter a SQL Select statement, and then run the Build process.

See [Chapter 6, "Creating Record Definitions," Creating SQL View and Dynamic View Select Statements, page 152.](#)

**Dynamic View**

Select to define a record definition that can be used like a view in pages and PeopleCode, but is not actually stored as a SQL view in the database.

Instead, the system uses the view text as a base for the SQL Select that is performed at runtime. Dynamic views can provide superior performance in some situations, such as search records and in PeopleCode Selects, because they are optimized more efficiently than normal SQL views.

---

**Note.** Keys for dynamic views should not be effective date fields.

The dynamic view should contain only fields whose names exactly match the fields comprising the SQL query.

---

**Derived/Work**

Select to define the record definition as a temporary workspace to use during online page processing. A derived or work record is not stored in the database, so you do not build it.

**SubRecord**

Select to define the record definition as a subrecord: a group of fields that is commonly used in multiple record definitions and that you can add to other record definitions. This way, you can change a group of fields in one place, as opposed to changing each record definition in which the group of fields is used.

**Query View**

Select to define the record definition as a view that is constructed using the PeopleSoft Query tool. Before you can create the view, PeopleSoft Application Designer prompts you to save the definition.

**Temporary Table**

Select to define the record definition as a temporary table. You can specify temporary images of the table on the PeopleTools Options page. Use temporary tables for running PeopleSoft Application Engine batch processes. Temporary tables can store specific data to update without risking the main application table.

**Non Standard SQL Table Name**

Specify the SQL table name that you are defining to override the standard convention of prefixing PS\_ to the record name.

**Build Sequence No.**

This field is available when the record type is a SQL or Query view. Set the order in which the view is to be created. The default value is 1 when the record or view is initially created. Views that must be created first can be set to 0, while views that you want created last can be set to any number from 1 to 99. The build sequence number is stored with the other details of the record or view in the database.

---

## Viewing Record Definitions

You can see four views of a record by selecting View from the main toolbar: Field Display, Use Display, Edits Display, and PeopleCode Display.

This section discusses how to:

- View basic field definitions.
- View key-related characteristics and default values.
- View editing options.
- View PeopleCode program types.
- Reorder fields.
- Size and sort columns.

### Common Elements Used in This Section

<b>Num</b> (number)	The number of the field in the order in which it is defined in the record.
<b>Field Name</b>	The name of the field in the SQL database.
<b>Type</b>	The data type of the field, such as character, number, or date.

### Viewing Basic Field Definitions

Field Display mode shows the basic field definition characteristics for fields in the record definition. Field definition characteristics are global: they affect all record definitions in which the field is used. To open the associated field definition, right-click it while in Field View and select View Definition.

<b>Len</b> (length)	Indicates the maximum length of the field, including decimal places.
<b>Format</b>	Notes special formatting for the field, such as mixed case, date, international phone number, or RawBinary.
<b>Short Name</b> and <b>Long Name</b>	Displays the short and long names of the field as users see them on pages.

## Viewing Key-Related Characteristics and Default Values

Use Display mode shows key-related characteristics and default values for fields that determine how fields are used in a record. The use characteristics might differ for fields that are used in more than one record definition. Double-click the field to access the Record Field Properties dialog box in which you define these parameters.

<b>Key</b>	<p>Indicates whether the field is a key to the record definition. Key fields are included in an index that is automatically created during the table build process.</p> <p>The word <i>Not</i> in this column indicates that this field cannot be used as a chart field. You can set this parameter only through PeopleCode using the SetDBFieldNotUsed function.</p> <p>See <i>PeopleTools 8.52: PeopleCode Language Reference</i>, "PeopleCode Built-in Functions," SetDBFieldNotUsed.</p>
<b>Order</b> (order)	Indicates the order of key fields in the index. You can configure this order from the Index dialog box. It is reflected in this column.
<b>Dir</b> (direction)	Indicates the order in which the key field indexes are created in the database: ascending or descending.
<b>CurC</b> (currency control)	The currency format for this field is controlled by the currency code in another field.
<b>Srch</b> (search)	Indicates a search key, that is, a field for which one is prompted from a search record dialog box.
<b>List</b>	Indicates a list box item, that is, whether the values for the field appear in the search record list box.
<b>Sys</b> (system)	Indicates that the system generates and maintains the field.
<b>Audt</b> (audit)	Specifies whether additions, changes, or deletions to data in this field are written to the standard PeopleTools Audit Tables (PSAUDIT and PSAUDITEXT).
<b>InAutoComp</b>	Indicates that the field is used as an autocomplete field and appears in the autocomplete window.
<b>EnAutoComp for Srch Record</b>	Specifies whether autocomplete is enabled in a search record.
<b>Default</b>	Represents the default value that is used to initialize the field.

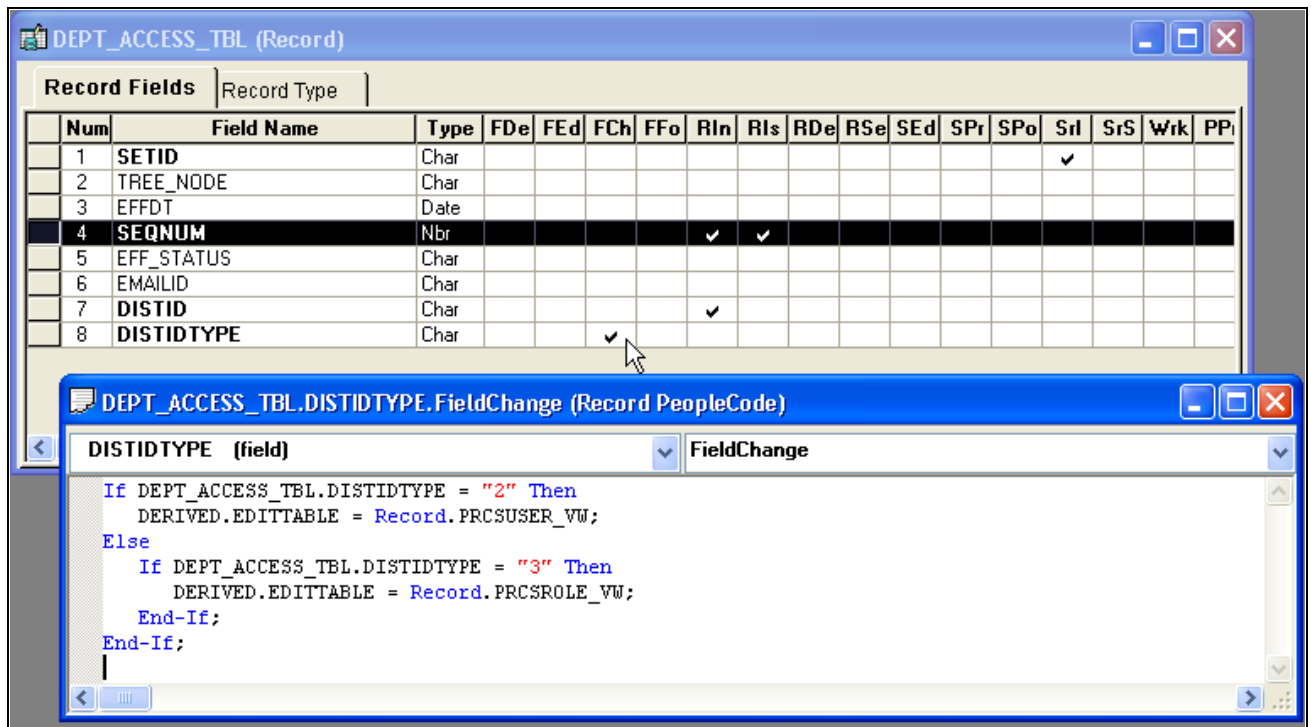
## Viewing Editing Options

Edits Display mode shows all editing options (edit as a validation rule) that are available for fields in a record. Edits on a field vary from one record definition to another. To define parameters, access the Record Field Properties dialog box by double-clicking the field.

<b>Req</b> (required)	Indicates whether the field is required. A user must enter a value before the record can be written to the database.
<b>Edit</b>	Specifies that the values for this field are validated against a table. The translate table stores the codes and translate values: <i>Y/N</i> (yes or no): Only two values, Y or N, are valid. <i>Prompt</i> : Values reside in a designated prompt table.
<b>Prompt Table</b>	Indicates the prompt table for a field. When a user clicks the prompt button or presses the ALT+5 key combination on a page, the system retrieves the values that are stored in this table.
<b>Set Control Field</b>	If you plan to use the Table Sharing feature to add an additional high-level key to identify common sets of values and handle exception values, add the name of the key here. The set control field determines which set of values appears, based on how you define table sharing.
<b>Rs Dt</b> (reasonable date)	Specifies whether a reasonable date test is performed on a date field. All date fields are automatically edited to ensure that you can enter only valid values. The reasonable date test warns if the date is outside a 30-day range before and after the current date. You can use this, among other tests, to guard against entering the wrong year in a date.
<b>Event</b>	Indicates whether any type of PeopleCode was added for this field. In addition, if PeopleCode was added, the field name will be boldface.

## Viewing PeopleCode Program Types

PeopleCode Display mode has a column for each PeopleCode program type. If a field contains a PeopleCode event, a check mark appears in the event type column. Double-click in a cell to launch the PeopleCode Editor. Note that fields containing PeopleCode are in bold in all display modes.



PeopleCode Display mode with PeopleCode Editor

## Reordering Fields

You can reorder the appearance of fields in a record definition by double-clicking the attribute name. For example, if you double-click Field Name, the fields appear in alphabetical order; double-clicking Num returns the fields to their numeric order. This action does not change the order of the fields in the actual record. The numbers that the fields are originally assigned remain the same. This fact is important when it comes to key fields.

To actually reorder the fields in the records, you must cut and paste or select a field and move it.

## Sizing and Sorting Columns

You can change column lengths in any of the displays by dragging them to the appropriate size with your cursor. For example, you might make the short name column smaller so that the entire long name for each field appears. The default sizing of all columns returns after you close the record and reopen it.

You can also sort the rows in columns by double-clicking the column heading. For example, double-clicking the Num column heading returns the list of record fields to the default sort order.



---

## Saving Record Definitions

You should save your work every time you define a new record definition. As soon as you add or change one element in the new definition, save your work and name the record. You cannot save a record definition until you make at least one change to the record definition by changing record properties, adding or deleting at least one field, or changing the record field properties.

You cannot add PeopleCode to a field until you save the record definition.

To save a new record definition, select File, Save or File, Save As. If you have not named the definition, the system prompts you to enter a record name. The system also prompts you to set the tablespace using the Change Space dialog box.

See [Chapter 7, "Administering Data," Setting the Tablespace, page 196.](#)

---

## Naming Record Definitions

Use these guidelines for record definition names:

- The name length can be up to 15 characters, with the exception of the Temporary Table type, which has a maximum length of 13 characters.
- The name must begin with a letter and can contain underscores to make it more readable.
- Avoid special characters, such as # or \$, which can cause problems in some database environments.

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33.](#)

### ***Record Naming Conventions***

To help identify the purpose of different types of record definitions, you should adopt these naming conventions for record definition names and use these suffixes:

<b><code>_TBL</code></b>	Identifies edit or prompt tables that contain data that is used for validation, as opposed to data that is maintained by the application. Prompt tables store commonly used values. They include, but are not limited to, control tables, which store company-wide values. For example, the location table (LOCATION_TBL) stores values for all operating locations in which your company does business; the country table (COUNTRY_TBL) stores values for all valid countries.
<b><code>_VW</code></b>	Identifies record definitions that are physically implemented by defining a SQL view.
<b><code>_DVW</code></b>	Identifies dynamic views.
<b><code>_WRK</code></b>	Identifies derived work records.

<b>_SBR</b>	Identifies subrecords.
<b>_QVW</b>	Identifies query views.
<b>_WL</b>	Identifies records as worklist record definitions.

In some cases, PeopleSoft also uses these prefixes to identify special types of record definitions:

<b>R_</b>	Identifies work record definitions for Structured Query Report reports. The remainder of the record name consists of the program or report ID.
<b>AUDIT_</b>	Identifies record definitions that store audit information for other record definitions in the database.
<b>WEBLIB_</b>	Identifies record definitions that store internet scripts. Internet scripts are generally located in FieldFormula PeopleCode events. You must grant access in the Security component for a WEBLIB record with an internet script before it can be run in a PeopleCode program.
<b>FUNCLIB_</b>	Identifies record definitions that contain written PeopleCode functions, as opposed to built-in functions. You can include these records in the component and call them as functions. These self-developed functions are generally located in FieldFormula events, and the records are usually derived.
<b>DERIVED_</b>	Identifies shared record definitions (across an application module or group) that have fields for PeopleCode events.

---

## Creating New Record Definitions

This section discusses how to create new record definitions.

To create a new record definition:

1. Select File, New.
2. Select Record.
3. Click OK.

The object workspace appears so that you can build a list of fields in a record definition.

4. Select the Record Type tab to define the type of record definition.

### See Also

Chapter 6, "Creating Record Definitions," Manipulating Fields in Record Definitions, page 125

---

## Opening Existing Record Definitions

Open a record definition in the same way that you open other definitions in PeopleSoft Application Designer. In the open dialog box, you can narrow the search by selecting a record type or project. You can also open an existing record from the project tree view by double-clicking a record name.

---

## Manipulating Fields in Record Definitions

This section provides an overview of fields in record definitions and discusses how to:

- Insert fields into records.
- Insert subrecords into records.

## Understanding Fields in Record Definitions

You create record definitions by adding field definitions to a new record definition or by cloning and modifying an existing record definition. You can add fields in any order and reorder them at any time. Locate keys at the top of the record definition, in order of importance.

Each field has basic attributes that are shared across all records that contain the field. These field properties include data type, field name, long name, short name, field length (or integer and decimal positions), formatting, help context number, and translate values. If you change any of these attributes for a field, the change affects *every* occurrence of the field in every record definition. If the change is not appropriate for every occurrence of this field, consider defining a new field instead.

---

**Note.** Changing a field name or length requires modification of the underlying SQL table, either by running the SQL Build or SQL Alter menu items or by a system administrator action. For example, if you change a field length for one record definition and 30 other record definitions contain the same field name, then you must build or alter 31 records. If the records are all within one project, you can run the build process on the project instead.

---

## Inserting Fields Into Records

You can insert a field into a record in one of several ways:

- Using the project workspace tree.

To insert a field into a record definition directly from the project workspace by using the drag-and-drop method:

1. Locate a field using the PeopleSoft Application Designer project workspace tree.
2. Drag the field from the project workspace tree to the object workspace.

The field and its attributes are automatically added to the list of fields in the record definition.

- Using the Insert menu (Insert, Field).

Use the Insert, Field menu selection to search for fields with selection criteria, and then add them to the record using the Insert button. You can also double-click a selected field definition and select Insert the Field Definition. The system inserts the fields below the selected field in the record definition.

- Dragging fields from existing records.

To drag a field from one record definition to another:

1. Open a record that contains the fields that you need.
2. Select the field.
3. Drag the field to a new record.

## Inserting Subrecords Into Records

Subrecords enable you to aggregate and configure a group of fields as a set, which you can then insert in multiple record definitions. All subrecord fields and configured properties propagate to the subsuming record definition.

You must define a subrecord before you can insert it into a record definition. You create and configure a subrecord as you do any record definition; however, you do not build subrecords because they are PeopleSoft structures, not standalone database structures. Select Insert, SubRecord to search, select, and insert subrecords into a record.

### ***Viewing a Subrecord***

Any open record in which a subrecord is inserted can have the subrecord expanded into the same record definition window by selecting View, Expand All Subrecords. After selecting this option, the parent record definition expands to show the fields in the subrecord. Subrecord fields are shaded.

When the subrecord is expanded, you cannot insert, cut, delete, paste, reorder, or sort fields. If you reordered the display of the fields, you cannot expand the subrecords.

The expanded subrecord fields are read-only, which means that you cannot change the properties of these fields. To access the properties, first open the subrecord. The shortcut to opening a subrecord is to click the expanded subrecord and select *View Definition*. This action opens a subrecord definition from which you can view the properties of the fields.

Collapse the subrecord by selecting View, Collapse All Subrecords.

### ***Nesting Subrecords***

Nested subrecords are fully supported to any level. Expanding a record toggles the record field list to show all of the fields from all levels of nesting. Changes to subrecords are immediately reflected in expanded records.

### ***Editing PeopleCode in Subrecords***

You can edit PeopleCode that is attached to a subrecord field by double-clicking the appropriate field as it appears in the expanded record. Any PeopleCode changes that you make apply to all records that contain that subrecord.

### ***Configuring Persist in Menu Properties***

All Persist in Menu properties that you explicitly define on record fields in subrecords propagate to all subsuming record definitions. When you configure custom persistent search fields, the Persist in Menu properties in both the subrecord and its subsuming records are counted toward the maximum number of persisted fields. Selecting the Persist in Menu property for record fields in a subrecord that is already being used by other record definitions displaces any lower order fields that, as a result of the subrecord reconfiguration, are beyond the five-field maximum.

See [Chapter 6, "Creating Record Definitions," Configuring Search Results Grid Columns in Persistent Searches, page 141.](#)

---

## **Setting Record Properties**

After you create a record definition, you will want to apply attributes so that the record functions. Access the dialog box by selecting Record Properties, and then select either the General tab or the Use tab. This section discusses how to:

- Set general properties.
- Set use properties.
- Create user-defined audit record definitions.

### **Setting General Properties**

Access the Record Properties, General tab.

<b>Description</b>	Enter a descriptive name for the record.
<b>Record Definition</b>	Enter appropriate comments in this area, including details about the record type, use, parent and child relationships, or other information that is important to other application designers.
<b>Owner ID</b>	Displays a list of applications with which this record is used. This list is helpful to identify the applications with which the record is associated during application development.
<b>Last Updated</b>	Displays the date and time of the last modification that was made to the record and the name of the user who made the modification.

### **Setting Use Properties**

Access the Record Properties, Use tab.

These properties control the way the record definition is used in the system.

Specify the field with which the system maps between the original key and the tablesets in the Set Control Field drop-down list box.

See [Chapter 5, "Planning Records, Control Tables, and TableSets," Defining Set Control Fields, page 109.](#)

## ***Record Relationship Settings***

<b>Parent Record</b>	If this is a subordinate or child record, specify the parent record.
<b>Related Language Record</b>	<p>Specify a related language record:</p> <ul style="list-style-type: none"> <li>• The master record definition, which contains the appropriate key and nonkey field definitions.</li> <li>• A clone of the master record definition, to which you add an additional key for language code.</li> </ul> <p>This is the related language record definition. This definition should contain only those nonkey fields for which contents vary by language. Link the two record definitions by specifying the name of the related language record definition in this field on the master record definition.</p> <p>At runtime, the system checks the user's language preference and retrieves the data value from the appropriate related language record definition.</p>
<b>Query Security Record</b>	Specify a query security record if you want secure access to a record using a security view.
<b>Analytic Delete Record</b>	Specify a delete record to use for optimization.
<b>Record Name</b>	Specify the user-defined audit record.

## ***Audit Options***

While you can audit individual fields at the field level, you might find it more efficient to have the system audit the entire row whenever a user adds, changes, or deletes information. For record-level audits, the system focuses on rows of data instead of specific fields. Consequently, a record-level audit writes a single row of audit data rather than writing multiple rows for each insert, change, or delete transaction at the field level.

<b>Add</b>	Inserts an audit table row whenever a new row is added to the table underlying the record definition.
<b>Change</b>	Inserts one or two audit table rows whenever a row is changed in the table underlying this record definition.
<b>Selective</b>	Inserts one or two audit table rows whenever a field that is also included in the record definition for the audit table is changed.
<b>Delete</b>	Inserts an audit table row whenever a row is deleted from the table underlying the record definition.

### **Record-Level Auto-Update Options**

These settings are used exclusively for mobile applications, enabling developers to display derived values on a mobile page. These settings are not available for subrecords.

---

**Important!** PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.

---

<b>System ID Field</b>	Set this field to assign a unique way to identify the record for mobile synchronization purposes. You must create a user-defined index for this field.
<b>Timestamp Field</b>	Set this field to automatically update the field with the date and time when there is a change to the record for mobile synchronization purposes.

### **Record Information**

Record Information contains options used during PeopleSoft upgrades.

<b>Tools Table</b>	Identifies SQL tables owned by PeopleSoft PeopleTools development, which is typically the same ownership as indicated by the PPLTOOLS project.
<b>Managed</b>	Identifies SQL tables owned by PeopleSoft PeopleTools development that contain information that is used by managed (cached) objects.
<b>Upgrade Job Not Needed</b>	Identifies SQL tables that <i>do not</i> require an upgrade job to be defined. This option applies only to "A to B" upgrades.

---

**Note.** The Append ALL (Dynamic Views) option applies to dynamic views. If it is set, then at runtime the system appends the WHERE clause generated at runtime to all the SELECT statements in a dynamic view with a UNION.

Also, if it is set, at runtime the system generates FILL as the table alias in the WHERE clause condition. For example, FILL.<record field name> LIKE '%O%'. You need to specify which table corresponds to the FILL in the dynamic view definition. For example, FROM table A FILL. You must specify the corresponding table because the same record field name may exist in two different records.

---

### **See Also**

PeopleSoft upgrade documentation for your installation.

## Creating User-Defined Audit Record Definitions

To audit at the record level, you must create a record definition and SQL table in which you store audit information. When you create a new audit record definition, name it with an *AUDIT\_* prefix. Some processes, such as the Employee ID Change and Employee ID Delete processes in the PeopleSoft Human Resources Management product line, change fields, such as EMPLID (employee identification). These processes do not affect record definitions that begin with the *AUDIT\_* prefix, leaving your audit data secure.

The easiest way to create an audit table is to open the record definition that you want to audit and save it as a new record definition, prefaced with *AUDIT\_*. Audit record definitions cannot contain key fields. Therefore, if you clone a record definition to create an audit record definition, you must remove all key attributes.

You should also use the audit-specific fields that are already defined for the PeopleTools audit tables (PSAUDIT and PSAUDITEXT), which the PeopleSoft application uses to track field-level audits. Place these audit fields at the top of the audit record definition, as you would keys. If you are creating several user-defined audit record definitions, you might consider creating and using a subrecord definition with these audit fields instead of adding them individually to each audit record definition.

This table lists audit field names and their purpose:

<i><b>Audit Field Name</b></i>	<i><b>Purpose</b></i>
AUDIT_OPRID	Identifies the user who caused the system to trigger the audits, either by performing an add, change, or delete to an audited field.
AUDIT_STAMP	Identifies the date and time the audit was triggered.
AUDIT_ACTN	<p>Indicates the type of action that the system audited. Possible actions include:</p> <ul style="list-style-type: none"> <li>• A: Row inserted.</li> <li>• D: Row deleted.</li> <li>• C: Row changed (updated), but no key fields changed. The system writes existing values to the audit table.</li> <li>• K: Row changed (updated), and at least one key field changed. The system writes existing values to the audit table.</li> <li>• N: Row changed (updated), and at least one key field changed. The system writes new values to the audit table.</li> </ul>
AUDIT_RECNAME	Identifies the name of the record definition that was audited.

In most cases, you should include AUDIT\_OPRID, AUDIT\_STAMP, and AUDIT\_ACTN. The AUDIT\_STAMP must be given the attribute AUTOUPDATE. You might also add AUDIT\_RECNAME if you are creating an audit table to audit more than one record definition.



---

**Note.** Select the Auto-Update check box in the Record Field Properties dialog box; otherwise, the audit record does not receive a date and time stamp.

---

---

## Setting Record Field Properties

This section provides overviews of record field properties and record key considerations, and discusses how to:

- Set record field use properties.
- Configure Search Results grid columns in persistent searches.
- Add from and through logic to a search page.
- Set record field edit properties.
- Move fields in the same record.
- Move fields to another record definition.
- Delete fields from a record definition.
- Rename record definitions.
- Delete record definitions.
- Print record definitions.
- Create SQL view and dynamic view Select statements.

## Understanding Record Field Properties

When you add a field to a record definition, you must define attributes for how the field is used in that record. These attributes include key settings, default values, table edits, and PeopleCode programs.

The properties of record fields are not shared among records; they are specific to a single record definition and are stored with the record. Therefore, even though you might add the same field to multiple records, each record stores a unique set of record field properties, while the primary field definition remains the same.

When you access field properties from within a record definition, you set the record field properties, not field definition properties or record definition properties.

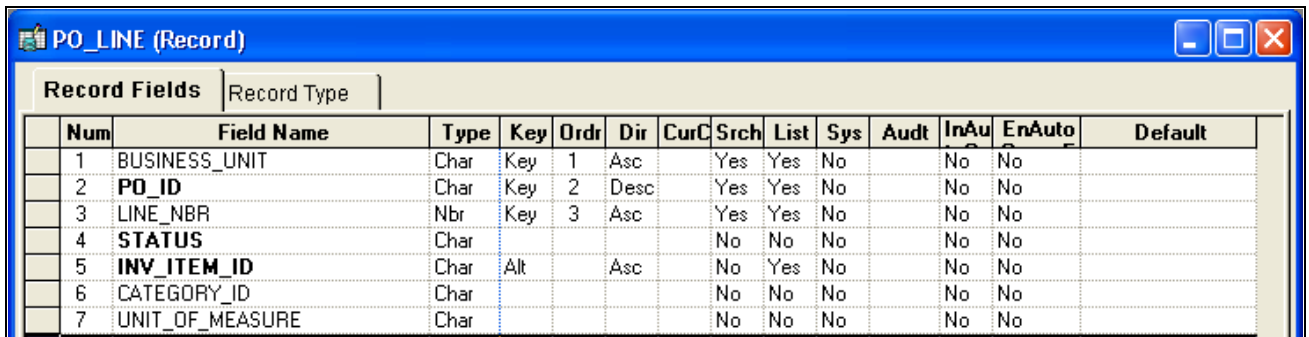
To edit record field properties from a record definition, select the record field, then select Edit, Record Field Properties from the PeopleSoft Application Designer menu. You can also double-click a record field to set the properties.

## Understanding Record Key Considerations

When creating key fields in your record definition, keep in mind several guidelines that are discussed in the following topics.

## Records With Multiple Keys

You can specify more than one field as a key to a record definition; that is, a record definition can have a compound key. This example shows a compound key structure:



The screenshot shows a window titled "PO\_LINE (Record)" with a tab labeled "Record Fields". Below the tab is a table with the following columns: Num, Field Name, Type, Key, Ord, Dir, CurC, Srch, List, Sys, Audt, InAu, EnAuto, and Default. The table contains seven rows of field definitions.

Num	Field Name	Type	Key	Ord	Dir	CurC	Srch	List	Sys	Audt	InAu	EnAuto	Default
1	BUSINESS_UNIT	Char	Key	1	Asc		Yes	Yes	No		No	No	
2	PO_ID	Char	Key	2	Desc		Yes	Yes	No		No	No	
3	LINE_NBR	Nbr	Key	3	Asc		Yes	Yes	No		No	No	
4	STATUS	Char					No	No	No		No	No	
5	INV_ITEM_ID	Char	Alt		Asc		No	Yes	No		No	No	
6	CATEGORY_ID	Char					No	No	No		No	No	
7	UNIT_OF_MEASURE	Char					No	No	No		No	No	

Example of a compound key structure in the PO\_LINE record definition

The keys to the PO\_LINE record definition are BUSINESS\_UNIT, PO\_ID, and LINE\_NBR. Therefore, each purchase order must be unique to a business unit, and each line item in a purchase order must have a unique number.

## Keys in Parent and Child Tables

In some cases, you have a field in a table for which you want to allow multiple occurrences, in which case you create a subordinate or child table. For example, for employee reviews, an employee can be reviewed for performance in multiple categories—organization skills, interpersonal skills, and so on. These categories and ratings are stored in a separate child table, EE\_REVIEW\_RT, which is directly related to REVIEW\_DT, the parent table that stores information about employee reviews.

The keys that you establish in a parent record definition determine which keys are required in child record definitions. The child must have the same keys as the parent, plus one or more keys that uniquely identify each row:

Item	Parent Table	Child Table
Record Definition	REVIEW_DT	EE_REVIEW_RT
Key Fields	EMPLID REVIEW_DT	EMPLID REVIEW_DT CATEGORY

Most record definitions have either one primary key, or multiple keys that comprise independent, or parent and child key combinations. Exceptions exist, however, such as record definitions without keys and duplicate order keys.

## Records Without Keys

Some record definitions, such as `INSTALLATION`, do not require keys because only one row of data exists in the table. Whereas a table normally has keys to help distinguish between multiple occurrences of data, when a table has only one row of data you do not need to distinguish one row from another. Another primary use for keys is to build database indexes to rows; because each table has only one row, the record does not need indexes. This example shows a record definition with no defined key structure:

Num	Field Name	Type	Key	Ord	Dir	Cur	Srch	List	Sys	Audt	InAu	EnAuto	Default
1	COMPANY	Char					No	No	No		No	No	
2	MIN_STD_HRS	Nbr					No	No	No		No	No	
3	MAX_STD_HRS	Nbr					No	No	No		No	No	
4	STD_HRS_DEFAULT	Nbr					No	No	No		No	No	'40'
5	TEMP_SSN_MASK	Char					No	No	No		No	No	'999'
6	COMMIT_AFTER	Nbr					No	No	No		No	No	'300'
7	POSITION_MGMT	Char					No	No	No		No	No	'N'
8	COUNTRY	Char					No	No	No		No	No	
9	COMP_FREQUENCY	Char					No	No	No		No	No	'M'
10	EMPLID_LAST_EMPL	Nbr					No	No	No		No	No	
11	EMPLID_LAST_APPL	Nbr					No	No	No		No	No	
12	NON_EMPLOYEE_LAST	Nbr					No	No	No		No	No	

`INSTALLATION` table: example of record definition with no defined key structure

## Records With Duplicate Order Keys

Occasionally, you might encounter situations in which a unique identifier for each row does not exist. *Duplicate order keys* are a way of ordering data in the table when duplicate values are allowed.

For example, in the `EDUCATN` table, PeopleSoft developers anticipated that an employee can receive two degrees of the same type on the same date. For example, though rare, Simon Schumaker could receive two honorary degrees in computer science on the same day. Because no unique identifier exists—the employee ID, date earned, and degree are all the same—the user must maintain the data differently. The key, instead of defining a unique row, defines a group of rows. In that group, you must determine the order in which you want to display the information.

In the `EDUCATN` record definition, three keys together determine how information is stored and retrieved. `EMPLID` identifies the group of rows; and the placement of the duplicate order keys, `DT_EARNED` and `DEGREE`, instructs the system to order rows in the group first by date and then by degree:

<i>EMPLID</i>	<i>DT_EARNED</i>	<i>DEGREE</i>
8001	June 1, 1992	HON
	June 1, 1992	HON

The system concatenates or joins keys when it checks for uniqueness. In this case, the system accepts duplicate entries in the DEGREE and DT\_EARNED fields because they are part of a set that is identified by EMPLID.

---

**Note.** Records with duplicate order keys should not have related language tables associated to them. If the record requires a related language record, you must remove the order keys. Do not create a related language record with a duplicate order key.

---

### **Ordering Keys**

The position of keys relative to one another is critical in a record definition; always list them in order of importance. The primary key must be the first field in the record, followed by the next most important key, and so on.

Key fields are followed by duplicate order keys, in order of importance, and then by fields that are not keys. You can scatter alternate key fields anywhere among regular fields, in any order. They do not need to be grouped.

Key, Duplicate Order Key, and Alternate Search Key are mutually exclusive.

---

**Note.** The display of key fields in record definition views does not always match the order in the record. Always check the Num column to see the actual order.

Each alternate search key that you configure creates a database index when the SQL creates the table. While database indexes are important, they consume disk space and, when the system must support the alternate key, processing time. Therefore, do not add alternate search keys unless you really need them.

For both records and subrecords, you can only add 10 alternate search keys.

---

## **Setting Record Field Use Properties**

As you add fields to a record definition, you must decide which fields uniquely identify each row; these become the record keys. The nature of the data that you are storing should naturally determine the keys for the information in the database.

The record field properties options that are available for field types might vary slightly; for example, some of the properties that you assign might be relevant only to a character field in a record and not to a date or time field. However, all of the options are described in this section. This example shows the available record field properties options:

The image shows the 'Record Field Properties' dialog box with the 'Use' tab selected. The 'Field Name' is 'COMPANY'. The 'Keys' section on the left contains several checkboxes: 'Key', 'Duplicate Order Key', 'Alternate Search Key', 'Descending Key', 'Search Key', 'Search Edit', 'List Box Item', 'From Search Field', 'Through Search Field', 'Default Search Field', 'Disable Advanced Search Options', and 'Allow Search Events for Prompt Dialogs'. The 'Audit' section contains 'Field Add', 'Field Change', and 'Field Delete'. The 'System Maintained' and 'Auto-Update' checkboxes are at the bottom left. On the right, the 'Record Field label ID' is set to '\*\*\* Use Default Label \*\*\*'. The 'Default Value' section has fields for 'Constant', 'Record Name', and 'Field Name'. The 'Default Page Control' is set to 'System Default'. The 'Autocomplete Configuration' section has an unchecked checkbox for 'Enable Autocomplete when used in Search Record'. The 'Persist in Menu Configuration' section has an unchecked checkbox for 'Persist in Menu'. 'OK' and 'Cancel' buttons are at the bottom right.

Record Field Properties dialog box: Use tab

## Keys

### Key

Select to identify the field as the search criteria that uniquely identifies each row. You cannot have duplicate values for primary keys. For example, EMPLID is the only key to the PERSONAL\_DATA record definition. Therefore, EMPLID must be a unique value for each employee, and only one PERSONAL\_DATA row can exist per employee.

### Duplicate Order Key

Select to indicate that duplicate values can occur. The order in which you place duplicate order keys in a field list determines the order in which the system retrieves duplicate keys.

<b>Alternate Search Key</b>	<p>Select to make this field part of the search structure when accessing your application data. Duplicate values are allowed in an alternate search key field.</p> <p>If you define a field as an alternate search key in a search record or prompt record, the system includes the field on the dynamically generated search page or prompt lookup dialog box that you use to enter search criteria.</p>
<b>Descending Key</b>	<hr/> <p><b>Note.</b> This option no longer produces an index with descending order.</p> <hr/> <p>Select to identify the field as descending if you want rows of data to be retrieved in reverse alphanumeric order (for example, 3, 2, 1). If you deselect this check box, the key is ascending, meaning that rows are stored and retrieved in alphanumeric order (for example, 1, 2, 3). This option applies only to a field that is specified as a key, duplicate order key, or alternate search key. PeopleSoft applications use descending order primarily for effective date fields; most character keys are ascending.</p>
<b>Search Key</b>	<p>Select to make the field available for entering search criteria on the basic and advanced search pages, the lookup prompt pages, and as a column that is stored in the menu structure for persistent searches. A search key is valid only for keys and should be used only in search and prompt records. If you select this check box, the system automatically selects the List Box Item check box.</p>
<b>Search Edit</b>	<p>Enabled only if Search Key is selected. Select to enforce the required property and table edits on the search page. It also enforces these edits under circumstances in which the search page would be bypassed. When this option is enabled, the user no longer has the ability to perform partial searches on this field.</p>

**List Box Item**

Select to include the field as a column in the search results grid on:

- Standard component search pages. The system includes *all* fields designated as list box items in the search results list.
- Prompt lookup pages. The system includes *all* fields designated as list box items in the search results list.
- Recent (persistent) search results pages. The system includes *no more than five* list box items in the search results list.

---

**Note.** This behavior is the default. To explicitly define which fields appear in the recent (persistent) search results, you must set other configuration options.

---

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Configuring the Portal Environment," Configuring Caching.

See [Chapter 6, "Creating Record Definitions," Configuring Search Results Grid Columns in Persistent Searches, page 141.](#)

If the field has values in the translate table and you designate it as a list box item, the list box automatically shows the translated value instead of the code.

If you select this check box, the system enables but does not select the Persist in Menu check box.

**From Search Field and Through Search Field**

Select for fields that are used as search records. If you select From Search Field, the displayed list contains rows in which the field value is greater than or equal to a value that the user enters. If you select Through Search Field, the displayed list contains rows in which the field value is less than or equal to a value that the user enters.

If you do not want these fields to be in a search list box, clear the List Box Item check box, even if the field is an alternate search key.

**Default Search Field**

Select to control which field name appears in the Search By drop-down list box on the basic search page.

**Disable Advanced Search Options**

Select to prevent runtime search pages from displaying advanced search features for this field.

**Allow Search Events for Prompt Dialogs**

Select to enable the SearchSave and SearchInit PeopleCode events on a prompt dialog for this field. The default value is *Cleared*.

---

**Note.** Limit your use of these events in this context. The amount and complexity of the PeopleCode affects the speed of data retrieval.

---

**Audit**

Audit options apply only to data that is manipulated on a normal PeopleSoft application page through component processing. These options do *not* apply to data that is added using SQLExec in PeopleCode or by some other means.

PSAUDIT and PSAUDITEXT are the standard PeopleTools audit tables for field-level audits. Both audit tables log identical information such as the user ID, time of change, and so on as well as the values of audited fields both before (OLD) and after (NEW) any changes.

Note these differences in the field properties of the two audit tables:

- In the PSAUDIT table, the field length of the OLDVALUE and NEWVALUE fields is 65 characters. Field values that are longer than 65 characters are truncated.
- In the PSAUDITEXT table, the field length of the PTOLDVALUE and PTNEWVALUE fields is 254 characters.

---

**Note.** If you are auditing a field with a field length greater than 65 characters, you should query PSAUDITEXT to ensure that you retrieve the entire field value and not a truncated subset.

---

To audit certain fields, regardless of the record definitions in which they are included, you want to control when the fields are audited. To audit several fields in a record definition, consider specifying audits at the record definition level.

<b>Field Add</b>	Audits this field whenever a new row of data is added.
<b>Field Change</b>	Audits this field whenever the content changes.
<b>Field Delete</b>	Audits this field whenever a row of data is deleted.
<b>System Maintained</b>	Select to indicate that the field value is system-generated. This option is for documentation purposes only and does not affect processing.
<b>Auto-Update</b>	Select to have the field updated with the server's current date and time whenever a user creates or updates a row. Any user entries—even if permitted on a page—are overwritten by the server time.
<b>Currency Control Field</b>	Enables the user to specify where to find the currency code that is used to display the currency symbol, decimal digits, and scale of a number field. To use this option, the multicurrency option must be set. You might also want to enable the current display when you define the field on the page.

### ***Record Field Label ID***

Specify which label, long name or short name, to use for the record. The default value is *\*\*\*Use Default Label\*\*\**. This setting enables you to change labels of record fields dynamically anytime that the default label on the field definition is changed.

For example, a field definition has three labels:

- Label1 (Long Name1, Short Name1), marked as default.
- Label2 (Long Name2, Short Name2).
- Label3 (Long Name3, Short Name3).



When the label ID in a record field is set to *Use Default Label*, initially the long name and short name are Long Name1 and Short Name1. If the default label is changed to Label3 in the field definition, then the long name and short name automatically become Long Name3 and Short Name3.

### **Default Value**

Select the most commonly used value as the default. You can always enter a different value if the default is inappropriate. The more defaults that you provide, the more data entry time you will save your users.

For a field, you can enter a default value as a system variable or as a combination of record and field names.

#### **Constant**

Specify a default value.

This value is case-sensitive. If the record has translate values for this field, the values appear in a drop-down list.

You can specify a system variable only when you want to display the value of the current date or time. The accepted system variables to be used as constants are:

- %DATE for DATE, TIME, or DATETIME fields.
- %TIME for TIME fields.
- %DATETIME for DATETIME fields.
- %CLIENTDATE for DATE, TIME, or DATETIME fields. Note that in this case %CLIENTDATE is only relevant for a DATE field. Otherwise it behaves the same as %DATE.

**Record Name and Field Name** Enter the record and field names of the default value.

#### **Default Page Control**

Specify the default appearance of a field as it appears on the page that corresponds with the record field that you are creating.

### **Autocomplete Configuration**

#### **Enable Autocomplete when used in Search Record**

Select to make this field appear as a column in the autocomplete list when the containing record is used as a search record.

This field is enabled for character, number, and signed number fields if the following conditions are met:

- The Default Page Control field value is *Edit Box* or *System Default*.
- The field is a search key or an alternate search key.

## ***Persist in Menu Configuration***

### **Persist in Menu**

Select to explicitly include the field as a column in the search results grid of a *persistent* search; select *only* if the field does not appear in the search results grid by default.

The Search Key and List Box Item properties determine which columns appear in the search results grid by default.

You must enable recent search results caching at the system level in the web profile for this property to take effect.

See [Chapter 6, "Creating Record Definitions," Keys, page 135.](#)

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Configuring the Portal Environment," Configuring Caching.

---

**Important!** If you select this check box for *any* record field in a search record, then the search results grid uses this property—not the list box item property—to determine *all* fields that appear in the grid. Consequently, you must select the Persist in Menu check box for each record field that you want to include in the search results grid.

---

See [Chapter 6, "Creating Record Definitions," Configuring Search Results Grid Columns in Persistent Searches, page 141.](#)

## ***Time Zone and Related Date Fields***

These fields are enabled only if the current field is a time or date and time field. They determine whether the field is displayed or entered in a specified time zone.

### **Specified Time Zone**

Times are always stored in a database base time zone, but when you place a time field on a page, you can display the time in the base time zone or another time zone.

If you deselect this check box, the time appears in the database base time zone. If you have users in multiple time zones, you can reduce confusion by showing the time zone along with the time.

If you select this check box, the system converts the time according to the time zone specified in Time Zone Control field. This control field must be a field in the current record. Set an appropriate default value for the time zone control field.

### **Date Control Field**

Select which related date field in the current record stores the calendar date to which this field should be adjusted.

Typically, you want to make a date field a descending key so that the row with the latest and most current time appears first. If you want the default value to be the current system date, enter the value %date (or %time) as the constant in the Default Values group box.

In general, each Time record field that is related to the Time Zone field should also be related to a Date field in that same record. Otherwise, the time data that appears or is saved may not properly reflect the appropriate state of daylight savings in that time zone. Furthermore, each Time record field should be associated with a *different* Date record field.

For applications where two Time record fields need to share a common Date record field (as in, records with a start time and an end time *always* occurring on the same date), you can relate each Time record field to the same Date record field. However, keep in mind that the Date record field will only consider itself associated with *one* Time record field—the related Time record field that has the earliest position in the record field ordering for that record.

## Configuring Search Results Grid Columns in Persistent Searches

The Persist in Menu property enables you to override the default fields that appear on the persistent search page and explicitly determine which fields the system includes in the search results grid. If you select the Persist in Menu property for *any* record field in the search record, then you must select the same property for *each* field, including search key fields, that you want to appear as columns in the search results grid.

---

**Note.** This property, when selected, controls the appearance of the persistent search results grid. Search key fields remain in the menu structure but are hidden from view.

---

To explicitly define the fields that appear in the results grid for a recent (persistent) search:

1. Open the search record.
2. Double-click the field you want to include in the results grid and select the List Box Item check box to enable the Persist in Menu check box.
3. Select the Persist in Menu check box for the field.
4. Click the OK button.
5. Double-click each additional field that you want to include and repeat Steps 2 and 3.
6. Save the record.

---

**Note.** When you save the record definition, PeopleSoft Application Designer alerts you if your custom configuration exceeds the maximum number of fields that can appear in the search results grid of a persistent search page. This alert is informational.

---

## Adding From and Through Logic to a Search Page

Refer to the following example when applying from and through logic to a search page:

CNT_CLAUSE_TBL (Record)							
Record Fields				Record Type			
	Num	Field Name	Type	Len	Format	Short Name	Long Name
	1	CONTRACT_CLAUSE	Char	3	Upper	Clause	Contract Clause
	2	EFFDT	Date	10		Eff Date	Effective Date
	3	EFF_STATUS	Char	1	Upper	Status	Status as of Effective D
	4	DESCR	Char	30	Mixed	Descr	Description
	5	DESCRSHORT	Char	10	Mixed	Short Desc	Short Description
	6	DESCR254	Char	254	Mixed	Long Descr	Long Description
	7	COMMENTS	Long	0		Comment	Comment

Example of a CNT\_CLAUSE\_TBL record

To apply the from and through logic to a search page:

1. Determine which component you want to change.

You must modify the search record for a particular page, so you must first identify the appropriate component.

2. Determine on which field to use the from and through search logic.

Open the component in PeopleSoft Application Designer and examine the search record that is associated with the component. In the CNT\_CLAUSE\_TBL record, the search record is CNT\_CLAUSE\_TBL, and the field to which you want to apply the from and through search logic is CONTRACT\_CLAUSE.

3. Create a new view that contains the same fields as the original search record.

To use the from and through search logic, the search record *must be a view*. If the search record is already based on a view, you can modify the existing view instead of creating a new view.

4. Select the From Search Field property for the field.

In the new view, select the from search field and apply the following record field properties: Key, Search Key, and From Search Field (List Box Item is optional).

5. Create a new field with exactly the same attributes as the from field.

For example, if CONTRACT\_CLAUSE represents the field from which you want to search, then create a field called CONTRACT\_CLAUSE\_TO representing the field through which you want to search.

6. Insert the new field directly below the original field.

7. Select the Through Search Field property for the field.

Select the new through search field and apply the following record field properties: Key, Search Key, and Through Search Field only.

8. Update the SQL view text.

Update the SQL view Select statement in the new view to reflect the new column. That is, the view text should select the same field twice in a row (as shown in the example that follows this procedure) because the same field is used for the from logic and the through logic.

9. Save and build the new view.

10. Update the component properties.

You must update the search record so that the component uses the new view.

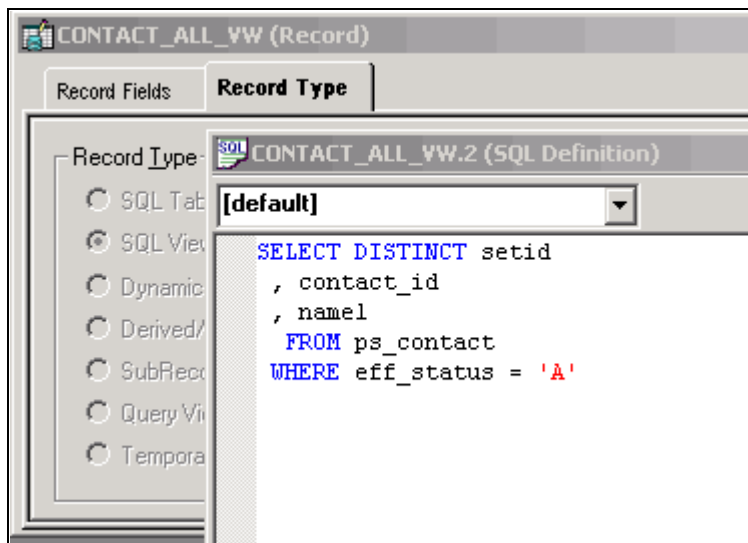
11. Test the new search page.

The search page should now include the new through field so that the user can specify a range of values for the field. Because you want to search on both fields, you must use an advanced search.

The results in the list box should reflect a range of values between the from and through fields that are specified. The results are inclusive: they include the minimum and maximum values that the user specified.

### **Example of Updating the SQL View Select Statement**

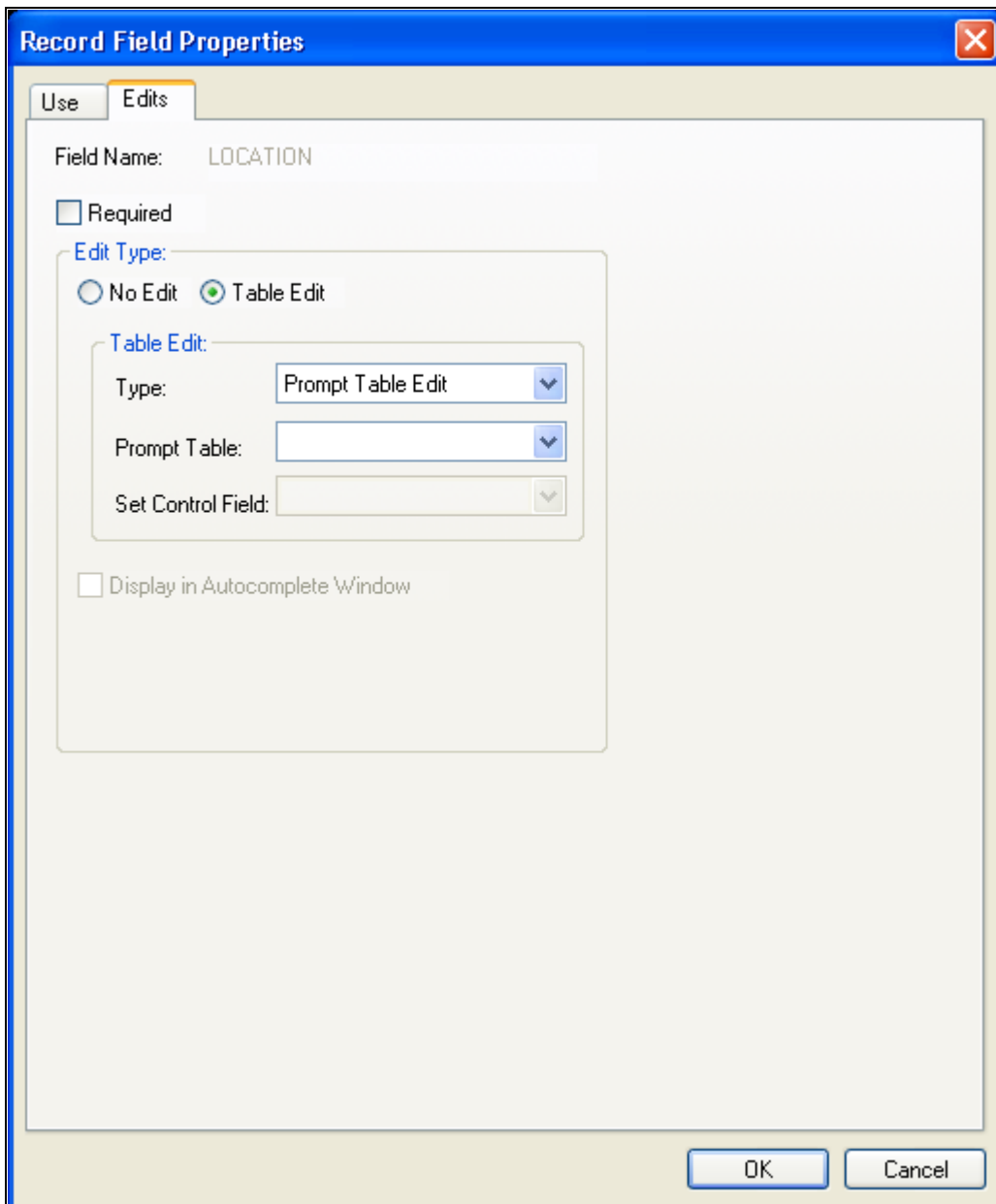
After you define a search field as a Through Search Field, you need to update the SQL view text, as shown in this example:



Example of updating the SQL view Select statement

## **Setting Record Field Edit Properties**

The Edits tab enables you to specify whether the system is to perform special edits or validations on a record field. You also name the record definition for the prompt table that stores the valid values, as shown in this example:



The image shows a screenshot of the 'Record Field Properties' dialog box, specifically the 'Edits' tab. The dialog has a blue title bar with the text 'Record Field Properties' and a close button (X) in the top right corner. Below the title bar, there are two tabs: 'Use' and 'Edits', with 'Edits' being the active tab. The main area of the dialog contains the following fields and options:

- Field Name:** A text box containing the value 'LOCATION'.
- Required:** A checkbox that is currently unchecked.
- Edit Type:** A section containing two radio buttons: 'No Edit' (unchecked) and 'Table Edit' (checked).
- Table Edit:** A sub-section containing three dropdown menus:
  - Type:** A dropdown menu with 'Prompt Table Edit' selected.
  - Prompt Table:** A dropdown menu that is currently empty.
  - Set Control Field:** A dropdown menu that is currently empty.
- Display in Autocomplete Window:** A checkbox that is currently unchecked.

At the bottom right of the dialog, there are two buttons: 'OK' and 'Cancel'.

Example of the Record Field Properties dialog box: Edits tab

### Required

Select if you do not want users to skip a field or leave it blank. Users are unable to save their work until they complete all of the required fields on a page.

---

**Note.** The system only checks required fields when the page field is changed or the record buffer is changed.

---

## **Edit Type**

### **No Edit**

Select if you do not want to edit the contents of this field against a table. This value is the default; it makes the options in the Table Edit group box unavailable.

### **Table Edit**

Select to edit the contents of the field against the values that are maintained in the specified table. When you select Table Edit, the Type field becomes available in the Table Edit group box.

### **Display in Autocomplete Window**

Select to make this field appear in the autocomplete list.

This property is enabled for date, datetime, time, character, number, and signed number fields if the following conditions are met:

- The field is a list box item.
- The field is not a search item.

## **Table Edit**

### **Type**

Select from the drop-down list box:

*Prompt Table with No Edit:* Provides users with a list of suggested values but does not edit the contents of the field against the prompt table. Users can enter any value. Selecting this option makes the Prompt Table field available.

*Prompt Table Edit:* Edits the contents of the field against the values that are maintained in the specified prompt table. Selecting this option activates the Prompt Table field. When you enter a prompt table name and exit the field by pressing the tab key, the Set Control Field is activated.

*Translate Table Edit:* Edits the contents of the field against the translate table. The translate table stores values for fields that must be validated but do not need individual tables of their own. If you select this option, the Prompt Table and Set Control Field fields become unavailable.

*Yes/No Table Edit:* Makes the values for this field Y (yes) and N (no) only. If you select this option, Prompt Table and Set Control Field fields become unavailable.

For this 1/0 table edit, 1=True and 0=False.

---

**Note.** The list of available table edit types is dependent on the field type.

If you add a field (other than with the paste action) to a record and the field name is already defined in the database with previously defined translate values, then the system automatically selects Table Edit and selects *Translate Table Edit* as the table type. For both *Translate Table Edit* and *Yes/No Table Edit*, you should follow PeopleSoft design standards, select Required, and specify a default value for the field in the Record Field Properties dialog box.

---

**Prompt Table**

Enter the name of the record definition that you want to use as the prompt table for this field.

If you want the prompt table to vary depending on the context of the field, indicate a field in the derived or work record (DERIVED) that contains the name of the prompt table at runtime. Then, in this field enter *%FieldName*.

The % is required, and indicates that you are referencing a derived or work record definition named DERIVED. *FieldName* is the name of the field in the DERIVED record definition.

**Set Control Field**

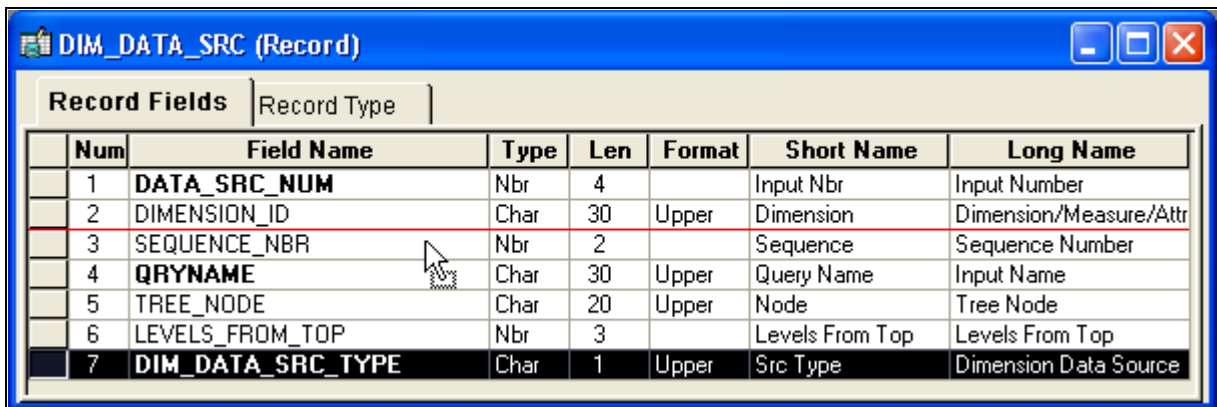
Select a set control field that overrides the set control field of the record definition specified in the prompt table. If you do not specify a name in this field, the default is the set control field of the record definition specified in the prompt table.

**Reasonable Date**

Select if you want the system to test the field value to determine whether it is within 30 days of the current date. If the date is out of range, a warning message appears when the user exits the field.

## Moving Fields in the Same Record

You can move fields in a record by dragging a field to another place in the open record definition. In addition, you can change the visible order of fields by clicking any of the grid headings. In this sample record definition, the DIM\_DATA\_SRC\_TYPE field is being moved to the location that is indicated by the red line:



Record Fields		Record Type					
	Num	Field Name	Type	Len	Format	Short Name	Long Name
	1	DATA_SRC_NUM	Nbr	4		Input Nbr	Input Number
	2	DIMENSION_ID	Char	30	Upper	Dimension	Dimension/Measure/Attr
	3	SEQUENCE_NBR	Nbr	2		Sequence	Sequence Number
	4	QRYNAME	Char	30	Upper	Query Name	Input Name
	5	TREE_NODE	Char	20	Upper	Node	Tree Node
	6	LEVELS_FROM_TOP	Nbr	3		Levels From Top	Levels From Top
	7	DIM_DATA_SRC_TYPE	Char	1	Upper	Src Type	Dimension Data Source

Moving field to new position

To move a field in the same record definition:

1. Select the field that you want to move.

Press the Ctrl and Shift keys to select multiple fields at once.



2. Drag the selected field number to the new position in the record definition window.

A red line between rows indicates the new placement options as you drag the field. The Num column automatically rennumbers the fields.

---

**Note.** The fields are reordered only in the visible display of the record definition, not in the actual table.

---

## Moving Fields to Another Record Definition

To move fields from one record definition to another, you can cut and paste the fields. You can also drag a field from the project workspace into a record definition in the object workspace or between open record definitions.

To move a field from one record definition to another:

1. Open the two records between which you want to move the field.
2. Select the field that you want to move.

Move multiple fields by using the Ctrl or Shift key.

3. Select Edit, Cut.

If this is the correct field to cut from the record, click Yes.

4. Select the new position for the field in the destination record.
5. Select Edit, Paste to paste the field into the new record definition.

---

**Note.** PeopleCode that is associated with fields is not carried over with cut and paste operations. The same is true for delete and undo with field deletions in records. RecordField attributes, such as key, search key, and so on, *are* retained.

---

## Deleting Fields From a Record Definition

To delete a field from a record definition:

1. Select the field that you want to remove and press the Delete key.

This action deletes the field completely and does not copy it to the clipboard, unlike a cut operation, which does copy it to the clipboard.

2. When the system prompts you to confirm the deletion, click Yes.

If you already created the underlying table for the record definition from which you are deleting the field, recreate the table or use the SQL Alter function to alter the table.

---

**Important!** If you delete a field from a record definition, you must also delete it from any pages on which it appears. When you delete a field, the system does not automatically delete references to the field in PeopleCode, so you must do this manually. To determine where the field is referenced in PeopleCode, use the Find Object References feature.

You can also reference the following two reports: "Fields and Records" (XRFFLRC) shows which records contain the field; "Fields Referenced by PeopleCode Programs" (XRFFLPC) shows PeopleCode that refers to the field in the record.

---

Modify or remove PeopleCode when you find references to the deleted field.

## Renaming Record Definitions

To rename a record definition:

1. Select File, Rename.

The Rename Definition dialog box appears.

2. Select *Record* from Definition Type.
3. Click Rename.
4. Select the record and click Rename.

A rectangular box appears around the name.

5. Enter the new name and press the Enterkey.

### ***Results of Renaming Record Definitions***

When you rename a record definition, the system automatically renames all references to it, including data in tables with columns named RECNAME where the data matches the record being renamed.

The only references that are *not* renamed are the text portion of SQL functions, such as SQLExec and Scroll Select, and the record names in the view text. To find the text portion of SQL functions in PeopleCode or record names in view text, select Edit, Find In to search for matching text.

If you already created the underlying tables for the record definition that you renamed, recreate that table. To preserve data in the tables, use the SQL Alter function to rename the database tables.

## Deleting Record Definitions

To delete a record definition:

1. Select File, Delete.
2. Select the record definition to delete.

3. Click Delete.

---

**Warning!** When you delete a record definition, the system automatically deletes any PeopleCode that is associated with the record.

---

4. Click Yes if you really want to delete the record definition.

---

**Note.** Notify your database administrator about which record definition you deleted so that the administrator can drop the underlying SQL table and its contents from the database.

---

## Printing Record Definitions

You can print your record definitions as references identifying all of the fields and their various attributes—any special use, edits, or PeopleCode that you applied.

To print a record definition, it must be open in the object workspace.

To print a record definition:

1. Select File, Print Setup to change any of the print record defaults.

The system retains your changes until you reset them again. The Print Setup dialog box appears.

## 2. Select printing options.

<b>Definition</b>	Print a picture of what the definition looks like.
<b>PeopleCode</b>	Select if you want the report to include a listing of any PeopleCode programs that are attached to each field, identifying the program type and listing all of the PeopleCode statements. Selected by default.
<b>Graphics</b>	Not used for record definition printing.
<b>Margins(mm)</b> [margins (millimeters)]	Set the distance (in millimeters) from the edge of the page to the left, right, top, and bottom edges of the page image or report. The defaults are: 20 (top), 5 (bottom), 5 (left), and 0 (right).
<b>Header</b>	Print a header at the top of the report indicating the date and time at which you printed the report and the database name, record name, version number, and page number. Selected by default.
<b>Footer</b>	Print a footer at the bottom of the report indicating the date and time at which you printed the report and the database name, record name, version number, and page number. Deselected by default.
<b>Border</b>	Print a border or box around a record definition report. To print reports faster, clear this check box so that the printer can print the report in character mode, rather than in graphics mode. Deselected by default.
<b>Border Space(mm)</b>	Insert a set amount of space between a graphical boarder around the record definition report and the margins of the report. The default value is 3.

## 3. Click OK when you are done, to close the Print Setup dialog box and save your settings.

## 4. Select File, Print.

**Record Definition Report**

The following table lists the columns in the record definition report and their contents:

<b>Column</b>	<b>Contents</b>
Field Name	The name of the field.
Type	The field type.
Length	The length of the field (not specified for long character fields).
Format	The field format.

<b>Column</b>	<b>Contents</b>
Long Name	The 30-character name of the field.
Short Name	The 15-character name of the field.
Key	Identifies key attributes (characters defined by position).
Req (required)	<i>Yes</i> indicates that the field is required.
TblEdt (table edit)	<i>Prompt</i> : Field values are edited against a specified prompt table. <i>Y/N</i> : The field uses the Yes/No Table. <i>Xlat</i> : The field has values in the translate table. Values (if printed) are listed below the field.
AU (auto-update)	<i>Yes</i> indicates that the auto-update option is enabled.
Dt (date)	<i>Yes</i> indicates that the reasonable date option is enabled.
PC (PeopleCode)	<i>Yes</i> indicates that the field contains PeopleCode. PeopleCode text (if printed) appears below the field.
Aud (audit)	<i>Yes</i> : The field audit flag is enabled. <i>A</i> : Audit add. <i>C</i> : Audit change. <i>D</i> : Audit delete.
Prompt Table	The name of the prompt table.
Default Value	Any default value or constant.

You might also see these additional reference lines below each field or at the end of the report:

<b>Reference</b>	<b>Description</b>
SQL View	Shows the SQL view Select statement for view-type record definitions. This appears at the top of the report.
System Maintained	Indicates that the System Maintained check box is selected. Because this option is for documentation purposes only, it doesn't warrant its own column heading.

<b>Reference</b>	<b>Description</b>
DbField Help Context: <i>nnn</i>	Indicates that the field has been assigned a field help context number to link it to a help file that describes how the field is used wherever it appears in the database.
RecField Help Context: <i>nnn</i>	Indicates that the field has been assigned to a record field help context number to link it to a help file that describes how the field is used only as it appears in this record definition.
Audit Record	Identifies a user-defined audit record. Lists the audit record name and the type of audit.
Set Control Field	Identifies a set control field that is designated for the record definition.
Related Language Record	Identifies a related language record that is designated for the record definition.
Query Security Record	Identifies views that are used to restrict query access to data that is stored in the table.
Parent Record Name	Identifies the hierarchical relationship of the record for query reporting.

Printing records with subrecords takes into account whether the record view is expanded or collapsed. If the record view is expanded, the subrecord fields are indented to the appropriate level of nesting.

## Creating SQL View and Dynamic View Select Statements

If you are creating a SQL view or dynamic view record definition, you must enter a SQL view Select statement to indicate which field values you want to join from which tables. The only difference between the standard view and dynamic view is that the dynamic view is not defined as a view to the database; it is stored on the client and run as a Select at runtime. Dynamic views avoid some constraints on views on some platforms.

<b>Non-Standard SQL Table Name</b>	Override the standard convention of prefixing each record name with <i>PS_</i> .
<b>Build Sequence No</b> (build sequence number)	Set the order in which the view is to be created. The default value is <i>1</i> when the record or view is initially created. Views that must be created first can be set to <i>0</i> , while views that you want created last can be set to any number from <i>1</i> to <i>99</i> . The build sequence number is stored with the other details of the record or view in the database.
<b>Click to open SQL Editor</b>	The view text is saved when the record is saved by selecting File, Save. The record must be saved first, before you open the SQL Editor.

As a general guideline, the SQL used for dynamic views should be as simple as possible because the runtime system appends WHERE clauses to the SQL definition to add key field lookup criteria and does not parse the dynamic view SQL definition to find correlation names. For example, we do not recommend the following format:

```
SELECT...FROM(SELECT * FROM...WHERE...)
```

Rather than having a WHERE clause in the subquery, one approach would be to create a static view that the dynamic view references. For example, that static view might be:

```
...
ABC_PRD_INS_VW as

SELECT B.SETID AS SETID
, A.BO_ID_CUST AS BO_ID_CUST
, A.INST_PROD_ID AS INST_PROD_ID
, B.PRODUCT_ID AS PRODUCT_ID
, B.DESCR AS DESCR
, A.SERIAL_ID AS SERIAL_ID
, A.INSTALLED_DATE AS INSTALLED_DATE
FROM PS_RF_INST_PROD A
, PS_PROD_ITEM B
, PS_RF_INST_PROD_ST C
, PS_PROD_PGRP_LNK D
WHERE A.SETID = B.SETID
AND A.PRODUCT_ID = B.PRODUCT_ID
AND A.SETID = C.SETID
AND A.INST_PROD_ID = C.INST_PROD_ID
AND C.INST_PROD_STATUS <> 'UNI'
AND A.SETID = D.SETID
AND A.PRODUCT_ID = D.PRODUCT_ID
AND D.PRODUCT_GROUP = '00'
```

Which is referenced by the dynamic view:

```
SELECT SETID
,BO_ID_CUST
,INST_PROD_ID
,PRODUCT_ID
,DESCR
,SERIAL_ID
,INSTALLED_DATE
FROM PS_ABC_PRD_INS_VW
WHERE SETID = :1
AND BO_ID_CUST = :2
AND INST_PROD_ID = :3
```

---

**Note.** The order of the columns in the Select statement must be identical to the field order in the corresponding record definition. Also, only certain types of meta-SQL statements can be used in view text.

---

**Note.** Related language records should not use dynamic views but should be limited to physical views and records.

---

### See Also

Chapter 7, "Administering Data," Using Cross-Platform Meta-SQL for Dynamic Views, page 194





## Chapter 7

# Administering Data

This chapter provides an overview of data administration and the Build process and discusses how to:

- Run the Build process.
- Create tables.
- Create indexes.
- Alter tables.
- Create triggers.
- Create views.
- Administer Data Definition Language (DDL).

---

## Understanding Data Administration and the Build Process

Using PeopleSoft Application Designer, you create several kinds of definitions that represent database components. For instance, field definitions represent table columns, and record definitions represent tables and views. From these field and record definitions, you can create tables, views, and indexes. The important concept to remember is that the definitions are just the blueprints for the actual, physical database components with which they are associated. In the context of the entire application development cycle, use the Build process to create SQL tables, views, triggers, and indexes after you define new fields and create the record definitions. After you build the SQL tables, you begin creating page definitions.

After you create and define your field and record definitions, you must create or build the underlying SQL tables that house the application data that your users enter online in your production environment. The Build process is the centerpiece of the data administration features in PeopleSoft Application Designer. The Build process uses DDL to construct a physical database component that is based on the associated record and field definitions that you created. Using the Build feature, you can create:

- Tables.
- Indexes.
- Views.
- Triggers.

You can also use the Build feature to alter existing tables if you change the record definition after the table already exists. Altering a table is useful, because it enables you to make changes without losing the application data that is already housed in the table. In general, the results of the build operation are written to a script file that a database administrator can run later. On some database platforms, you can run the SQL online, if you prefer, so that your changes are immediately reflected in the physical database. However, if you run the SQL immediately, you cannot review it to make sure that the table that was built truly meets all of your requirements.

Before you begin using the data administration tools and running a build process, make sure you:

- Review what DDL means for PeopleSoft applications.
- Grant build authority.

### ***Reviewing DDL***

DDL is the part of SQL that pertains to the creation of tables, indexes, views, triggers, and tablespaces. DDL is also the part of SQL that differs most between the various relational database platforms. Each database vendor provides different syntax and configuration options for creating and organizing tables and for optimizing performance. Because PeopleTools supports multiple database platforms, PeopleSoft developers designed a flexible way of specifying DDL that enables you to take advantage of each vendor's features. The basic components of the PeopleSoft DDL support include the following:

- DDL model definition: A complete set of the supported DDL statements for each database platform.  
Statements include Create Table, Create Tablespace, and Create Index. Each DDL model statement has substitution parameters that can be specified at the database level or overridden for individual records.
- Record DDL: Specify the DDL model substitution parameters for an individual record.
- Index DDL: Specify the DDL model substitution parameters for an index.
- Sizing sets: A way to maintain multiple versions of your DDL model statements for a database platform.

For example, you can use one sizing set during a development phase, when tables have only test data, and another during production, when tables have more data.

### ***Granting Build Authority***

PeopleSoft Security enables you to specify which users can build scripts, run scripts (Execute SQL now), maintain DDL, and so on. Access Security from the PeopleTools link in the menu.

### ***See Also***

*PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting PeopleTools Permissions

---

## **Running the Build Process**

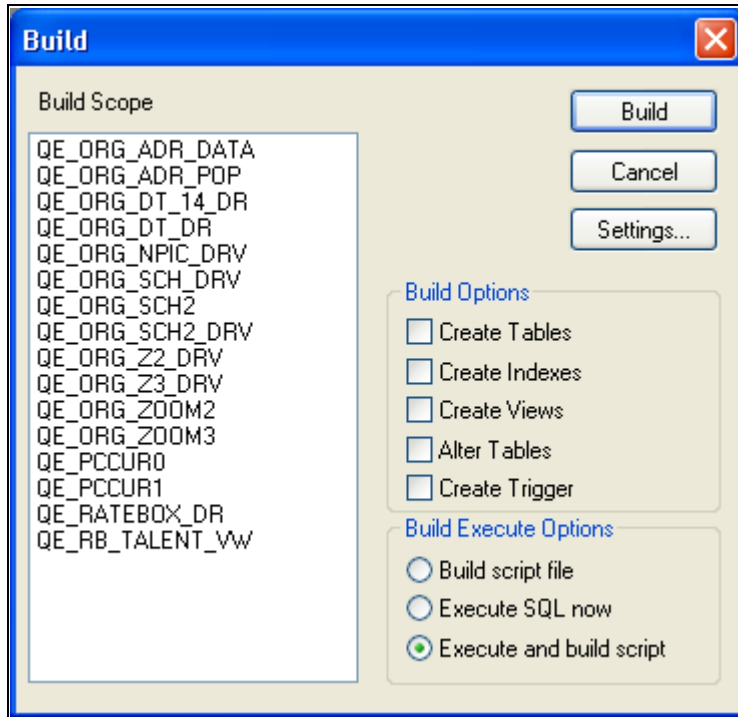
This section provides an overview of the Build menu and discusses how to:

- Select the build scope.

- Select build options and run the Build process.

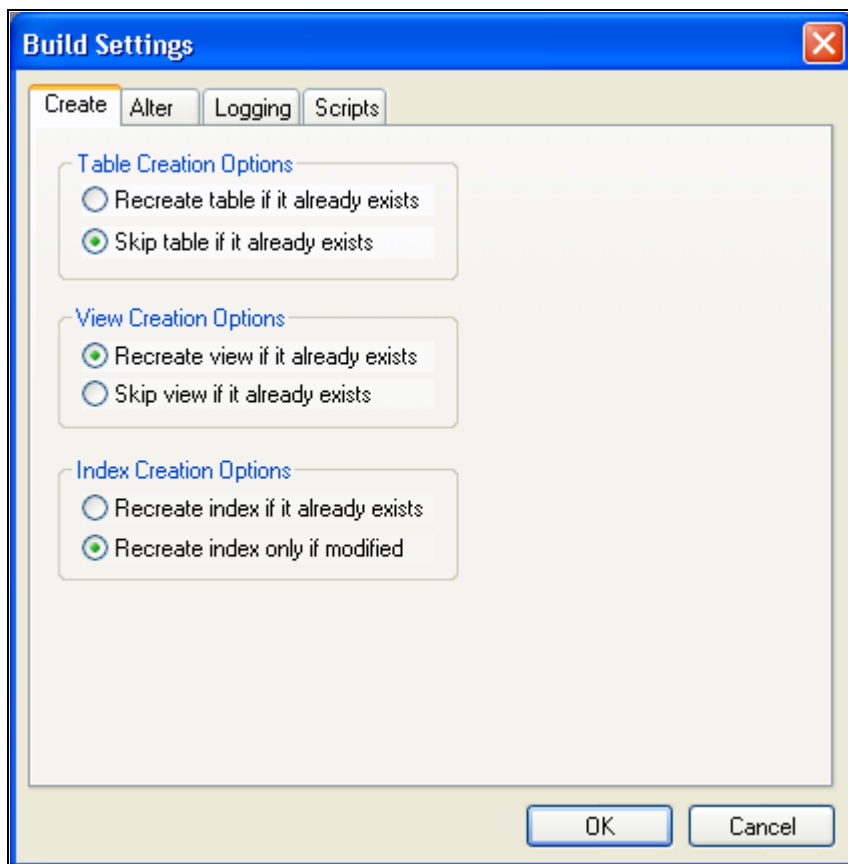
## Understanding the Build Menu

Access all of the commands and options that are associated with building tables, views, indexes, triggers, and altering tables from the Build menu. When you choose to build an item from the Build menu, the tasks for creating or running SQL scripts that define the underlying database components are included in the Build dialog box.



Build dialog box

The settings for the Build process in the Build Settings dialog box enable you to configure various aspects of the process. Depending on which build option you are running, your settings vary. The Build Settings that you select are stored on your workstation so that they can be reused during future PeopleSoft Application Designer sessions. Access the Build settings by selecting Settings from the Build menu in PeopleSoft Application Designer or click the Settings button in the Build dialog box.



Build Settings dialog box

When you select the Create Tables, Create Indexes, or Create Views options in the Build dialog box, specify the appropriate options on the Create tab of the Build Settings dialog box. For example, you might need to specify whether an existing table, view, or index is skipped or dropped and then recreated.

If you modify your record definitions, the tables in your production database should reflect those changes to maintain your data integrity. When you alter a table, select the appropriate options, such as the options in the Drop Column Options area on the Alter tab of the Build Settings dialog box.

You can monitor the Build process by reviewing the log files that it automatically generates. Keep in mind that the log file is *entirely* separate from the script file. The amount of information that the log file contains is up to you. You can set up logging so that all statuses (both good and bad) appear in the log, or you can specify that only errors or warnings appear in the log. Control the settings for the Build log file (such as the logging level and the location for the output log file) on the Logging tab of the Build Settings dialog box.

If you build a script file, then the Build process generates a script file that contains all of the CREATE and ALTER SQL statements so that you can review them before running the SQL through another SQL command processor. If the generated script file meets your requirements, a database administrator (DBA) can run the script later. The Build process can produce multiple scripts during a single run—one for each build option—depending on the script settings that you specify at runtime. For example, you can specify that the Build process generate a separate script for your tables, views, indexes, triggers, and alters, or you can have all of the SQL for each action contained in one script. Use the Scripts tab of the Build Settings dialog box to specify where the SQL script is written, whether you want multiple scripts generated for each definition type, and whether you want previous scripts overwritten.

## Selecting the Build Scope

The Build feature includes three different scopes that determine the size of your build project. Depending on what you actually want to build—one record or an entire project—you can determine the scope by selecting one of the scopes described in the following topics.

### **Current Definition**

Select Current Definition from the Build menu to build or alter the active record definition. *Active definition* means the definition that is currently active in the definition workspace.

### **Project**

Select Project from the Build menu to build or alter all of the records, indexes, triggers, and views in the current project—that is, all of the records that appear in the project workspace. You typically find numerous record definitions in the Build Scope list box.

To build all definitions in a database, create a new project and insert all of the appropriate records. For example, insert records of a particular type, such as views. If you create another record after the *all records* project is built, you must manually add the new record to the project.

You may also build a project using the command line and specific parameters. The prerequisites for using the command line to build a project are:

- The project exists with all objects.
- The build settings need to be set in the Windows registry.

---

**Note.** The Build Project process accesses every single record that is in the project, including work records, even if they are marked to be deleted.

---

### **Select Definitions Into Project**

In some situations, you might want to build a subset of the records in the project that is currently open. If you do not want to build an entire project or even the current project, you can build only the definitions that you select in the project workspace by pressing the ctrl key and selecting multiple records in the project workspace. After you select the appropriate records, right-click and select Build from the pop-up menu. Only the definitions that are selected in the project workspace appear in the Build Scope list box.

---

**Note.** You must invoke the Build process from the pop-up menu; you cannot build selected definitions from an open project using the Build menu.

---

## Selecting Build Options and Running the Build Process

Access the Build dialog box.

<b>Build Scope</b>	This list box displays all of the definitions that are included in the current Build process. The records that appear in this list box cannot be selected or deselected. If you must narrow the scope of your build, do so before accessing the Build dialog box.
<b>Build</b>	Click this button to start the Build process. While the Build process is running, the Build Progress dialog box appears, showing the current record being processed, the record number being processed in the series, and the total number of records that comprise the build. To stop a running Build process, click the Cancel button. When the Build process completes, the Cancel button is unavailable, and the Close button is the only valid button.
<b>Settings</b>	Click this button to open the Build Settings dialog box, where you can view or change your options.

---

**Note.** After the Build process begins, no runtime prompts appear until the process is complete. All events are written to the Build log.

---

### ***Build Options***

The Build Options group box enables you to specify what action you want to occur: Create Tables, Create Indexes, Create Views, Alter Tables, or Create Trigger. If you select Create Tables or Alter Tables, Create Indexes is automatically selected by default. Also, if you select Alter Tables, Create Indexes and Create Trigger are automatically selected by default.

---

**Note.** You can select both Alter Tables and Create Tables to run concurrently if the Skip table if it already exists option is enabled on the Create tab in the Build Settings dialog box. Otherwise, the system cannot determine whether you want to alter or recreate a table that already exists.

---

### ***Build Execute Options***

<b>Build script file</b>	Select this option to review and update the SQL before running the script. This method is the safest.
<b>Execute SQL now</b>	Select this option if you do not want to invoke another program to run the SQL; the SQL runs as part of the Build process. The disadvantage of running the SQL immediately is that you have no opportunity to review the SQL before it is committed to the database.
<b>Execute and build script</b>	Select this option to review the SQL that the Build process just ran.

To run the SQL script file that the Build process generates, you must use a third-party SQL interpreter, such as Oracle SQL\*Plus or Microsoft ISQL/W. Most database vendors include a native command processor that you can run on the client or the server.

**Note.** The Execute SQL now option is unavailable if you use the Alter Tables Build option. However, you can use the Execute and build script option when using Alter Tables. Selecting this option builds a script with the entire altered SQL. This script may help if errors are encountered during the alter process. However, if an alter process has multiple tables and an error occurs, the errors are reported to a log file and the process continues to the next table.

## Using Command Line Syntax to Build a Project

Use the same command line syntax as identified previously to build a project.

See [Chapter 19, "Using Command Line Parameters," Understanding PeopleSoft Application Designer Command Line Parameters, page 531.](#)

Use the command line statement for the project build to:

- Connect the project to the source database.
- Use the build settings from the Windows registry to generate the SQL script for the create or alter processes.

Before running the build command line, verify that:

- The system components of the project are created and loaded.
- The build options are set in the Windows registry.

### Available Parameters

This table lists all of the available parameters for the project build statement:

<i>Parameter</i>	<i>Description</i>	<i>Error Handling</i>
-HIDE	Use to hide the PeopleSoft Application Designer interface. Always use this parameter when performing a command line Copy process.	Required for all project command line processes.
-CT	Database type. Enter the type of the database to which you are connecting (MICROSOFT, ORACLE, SYBASE, and so on).	Required. If you omit this parameter, the last database type is taken from the registry. If it fails, the process stops and error messages are written to the log file.
-CS	Server name. Enter the name of the source database server for the database to which you are connecting.	Required for some database types. If you omit this parameter, the process stops and error messages are written to the log file.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
–CD	Database name. Enter the name of the source database to which you are connecting as you would enter it into the PeopleSoft Sign in dialog box.	Required. If you omit this parameter, the process stops and error messages are written to the log file.
–CO	User ID. Enter the PeopleSoft user ID that is needed to sign in to the source database.	Required. If you omit this parameter, the last database type is taken from the registry. If it fails, the process stops and error messages are written to the log file.
–CP	User password. Enter the password for the specified user ID for the source database.	Required. If you omit this parameter, a PeopleSoft Sign in dialog box appears for the user to enter a valid user password. If the password fails, the process stops and error messages are written to the log file.
–PJB	Project name. Enter the name of the project to be built. This project should be available in the database before starting the command line project build.	Required. The system uses this main parameter internally to decide whether the user can build a project. If you omit this parameter and if you supply all of the source database login parameters, the program only starts the application.

### ***Example***

For this example, assume that the:

- Project name is CJR1.
- Database type is Microsoft.
- Database name is CJR810G.
- User ID is PTDMO.
- Password is PTDMO.
- Path name of the SQL script file is c:\temp\psbuild.sql.
- Path name of the log file is c:\temp\psbuild.log in the Windows registry.

For this example, you would enter the following at the command line:

```
PSIDE.EXE –CT MICROSOFT –CD CJR810G –CO PTDMO –CP PTDMO –PJB CJR_PRJ
```



---

## Creating Tables

This section provides an overview of the Create Table process and discusses how to:

- Create a table.
- Specify Create options.
- Specify Logging options.
- Specify Script options.
- Confirm a table build.

## Understanding the Create Table Process

The Build process generates the appropriate SQL Create statements to build tables based on both your record definition and your platform. It prefaces each new application SQL table with *PS\_* to identify it as an application that was built using PeopleTools. For example:

`PS_ABSENCE_HIST`

---

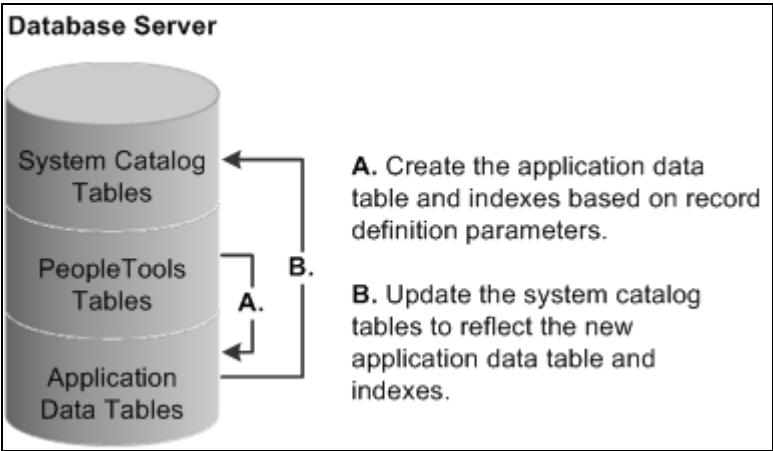
**Note.** PeopleTools does not preface table names with *PS\_* if you specified a custom name in the Non-Standard SQL Table Name edit box located on the Type tab of the Record Properties dialog box.

---

The Build process also creates the appropriate indexes. Because long-character fields operate more efficiently in some database environments when they are located at the end of a table, the Build process automatically positions long-character fields at the end of the table for each database environment.

The Create Table process creates a new application table based on parameters defined in the record definition. When a new table is created, the database management system updates the system catalog tables to reflect the attributes of the new table.

The Create Table process is illustrated by the following diagram:



Create Table process

After you create an updated SQL build script and run it, the PeopleTools tables and the system catalog tables are synchronized; and the record definition and the application data table are synchronized. The following table lists the steps of the Create Table process and the associated record definition parameters:

Create Table Process	Record Definition Parameter	Usage
Drop the table if it already exists.	NA	NA
Create the application data table.	Record definition name. Field names. Field type. Field length.	Table name (add <i>PS_</i> prefix). Column names. Column type. Column length.
Create indexes.	Key fields. Alternate search key fields. Custom indexes.	Unique index: ( _ ) Alternate index: 10 (0–9) per table. Custom index: (A–V) per table.

The following procedure covers all of the high-level steps that you need to successfully complete for the Create Table build process. Steps that involve a variety of options include links to the area where those options appear in PeopleBooks.

**Note.** You can have a maximum of 500 columns per table.

Creating a Table

To create a table:

1. Open the project for which you want to build SQL tables.
2. Select Build, Project.

3. From the Build menu, select the appropriate scope of your build.

See [Chapter 7, "Administering Data," Selecting the Build Scope, page 159.](#)

The Build dialog box appears.

4. Select the Create Tables in the Build Options group box.

PeopleSoft Application Designer automatically activates the Create Indexes option. If you are rebuild or recreate tables, then the indexes that the system uses to extract information from those tables must also be updated. Definitions to be built appear in the Build Scope list box, which does not allow you to edit, remove, or add definitions.

5. Select one of the Build Execute options.

The default option builds an SQL script file containing the commands to run the selected build options.

6. Click the Settings button in the Build dialog box to set user-defined defaults.

The Create tab in the Build Settings dialog box appears.

7. Select the appropriate settings on the Create tab.

See [Chapter 7, "Administering Data," Specifying Create Options, page 165.](#)

Recreating a table removes all data contained in the table as well as views or grants referencing that table.

8. Select the Logging tab.

9. Set the Logging Levels and associated options.

See [Chapter 7, "Administering Data," Specifying Logging Options, page 167.](#)

10. Select the Scripts tab.

11. Specify the script file options.

If you selected Execute SQL Now as your build option in the Build dialog box, you can skip the Scripts tab because you will run the SQL online. Settings on the Scripts tab are relevant only if you are building a SQL script to run later.

See [Chapter 7, "Administering Data," Specifying Script Options, page 171.](#)

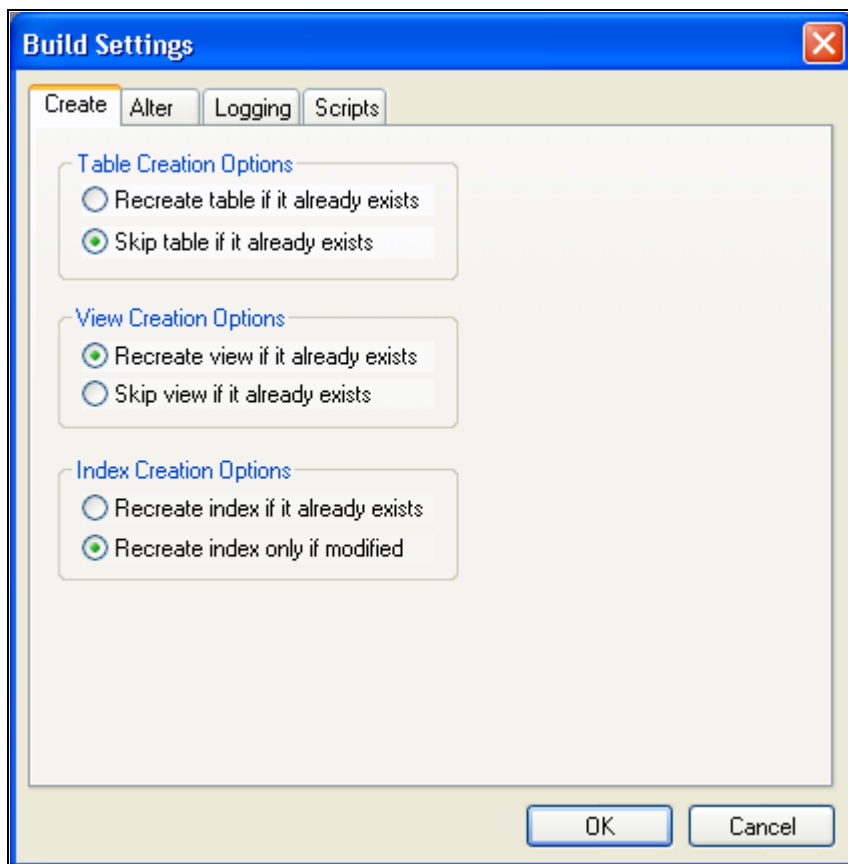
12. Click OK to close the Build Settings dialog box.

13. In the Build dialog box, click the Build button to run your build option.

The length of a Build process depends on the number of definitions that require building. Watch the Build Progress dialog box to monitor the Build process. When the process completes, check any errors listed in the log file.

## Specifying Create Options

Access the Build Settings dialog box and select the Create tab.



Build Settings dialog box: Create tab

### ***Table Creation Options***

These options determine when a table should and should not be created.

#### **Recreate table if it already exists**

Select to drop and recreate a table if it already exists. Select this option with extreme care because if data already resides in the table, it is also dropped. If you select this option, the Build process prompts you to confirm your intention before performing any destructive action. If you do not care about losing the data that resides in the table, then this option is faster than the Alter Tables option.

#### **Skip table if it already exists**

Select to create only newly defined tables. If you want to preserve the data that already resides in existing tables or if you are only interested in creating the tables that do not already exist, select this option.

### ***View Creation Options***

Similar settings are available for creating views that you use for creating tables.

<b>Recreate view if it already exists</b>	Because views are only logical views of data, you can safely select this option; you do not run the risk of losing physical data. Using this option ensures that all views in the project are rebuilt, whether they need to be or not.
<b>Skip view if it already exists</b>	If you are concerned only with creating views that do not already exist in the database, select this option. This option is useful if you want to run Build Project on a large project and only a subset of the views in the project must be created. This option consumes less time than recreating all of the views in the project.

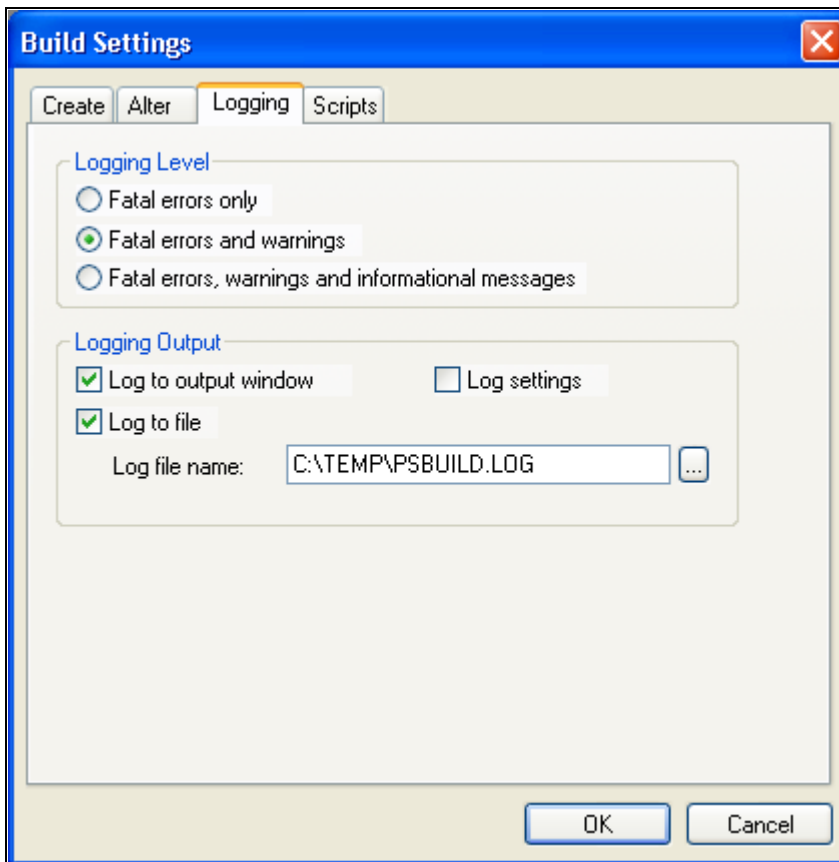
### ***Index Creation Options***

These options determine when an index should be recreated.

<b>Recreate index if it already exists</b>	Select to recreate the index for your tables, even if you already have one in place. This option recreates the index no matter what the situation.
<b>Recreate index only if modified</b>	Select to recreate the index only if the data in the indexes is being modified.

## **Specifying Logging Options**

Access the Build Settings dialog box and select the Logging tab.



Build Settings dialog box: Logging tab

### ***Logging Level***

Specify the level of detail of information that you want in your Build log output.

<b>Fatal errors only</b>	Select this option if you want to see only the operations that failed.
<b>Fatal errors and warnings</b>	Select this option to add warnings to the fatal errors.
<b>Fatal errors, warnings and informational messages</b>	Select this option to see all the definitions that processed successfully <i>and</i> all of the errors and warnings.

---

**Note.** These options are personal preferences and do not affect the actual build of your SQL tables, but they might help you track down potential failures. For example, if you are processing a large number of definitions, you will want to see only the operations that failed. Whereas, if you are processing a small number of definitions, you will want to see all of the successful operations, too.

---

### ***Logging Output***

Specify where and in what form you want to view your log output.

<b>Log to output window</b>	Select to write the same information that appears in the log file to the PeopleSoft Application Designer output window. Always be aware of the level that you select. If you expect a lot of detail (high logging level plus many records to process), you may want to copy the logging information to a file and print it later.
<b>Log to file</b>	Select to write the log to a file. If you select this check box, you can specify a location and name for your log file in the Log file name edit box. To search for a name, click the Browse button. Select a .LOG type file from the Build File Selection dialog box. If you do not select this check box, the edit box is unavailable for entry.
<b>Log settings</b>	Select to write all of the runtime options to the log. If you select this check box, you do not need to rerun a test when you have a problem. If you must report a problem with the Build process or troubleshoot, you need to know the settings that were active.

---

**Note.** You should select the Log to file check box and the Log to output window check box with a minimum logging level of fatal errors and warnings. In case you run into a problem, you already have the information you need to research the problem. Otherwise, you must run the process again with a higher logging level.

---

### **Example Log Results**

The following are samples of the type of information that you can expect to see in your log output that corresponds to the selections that you make on the Logging tab:

- **Fatal errors:**

```
SQL Build process began on 7/30/2003 at 8:46:46 PM for database
OM51U20
Error: BI_BILL_BY_F_VW - Unable to retrieve current DDL space name.
Alter processed failed. (76,47)
Error: BI_CHARGE_VW - Unable to retrieve current DDL space name.
Alter processed failed. (76,47)
Error: PS_BI_CYCLE_VW - alter failed due to an unknown column type
(UNDEFINED) found for field BILL_CYCLE_ID. (76,22)
Error: PS_BI_IVCLAYOUT_VW - alter failed due to an unknown column
type (UNDEFINED) found for field INVOICE_LAYOUT_ID. (76,22)
Error: CARINA - alter failed because SQL table does not exist. (76,7)
SQL Build process ended on 7/30/2003 at 8:54:34 PM.
2487 records processed, 11 errors, 0 warnings
SQL Build script for all processes written to file C:\TEMP\step38.SQL.
SQL Build log file written to C:\TEMP\step38.LOG
```

- **Warnings:**

```
SQL Build process began on 8/27/2003 at 4:00:32 PM for database DXD7L
Warning: ABSENCE_HIST2 - data would be truncated by altering
REASON. Record processed. (76,24)
```

- Fatal errors, warnings, and informational messages:

```
SQL Build process began on 8/27/2003 at 3:58:58 PM for database DXD7L
Warning: ABSENCE_HIST2 - data would be truncated by altering
REASON. Record processed. (76,24)
Error: ABSENCE_HIST2 - field PAID_UNPAID to be deleted has data
present. Record skipped. (76,23)
Informational: ACCOMPLISHMT_VW - view created and scripted
successfully. (76,18)
Informational: ACCT_AD2_INV_VW - view created and scripted
successfully. (76,18)
Informational: ACCT_ADJ_INV_VW - view created and scripted
successfully. (76,18)
Informational: ACCT_ISS_INV_VW - view created and scripted
successfully. (76,18)
Informational: ACCT_STK_INV_VW - view created and scripted
successfully. (76,18)
Informational: ACCT_TYPE_VW - view created and scripted
successfully. (76,18)
Informational: ACTION_SP_VW - view created and scripted
successfully. (76,18)
Informational: ACTION_XLAT_VW - view created and scripted
successfully. (76,18)
Informational: ADJUST_INV2_VW - view created and scripted
successfully. (76,18)
Informational: ADJUST_INV_VW - view created and scripted
successfully. (76,18)
11 records processed, 1 errors, 1 warnings
SQL Build script for all processes written to file
C:\TEMP\PSBUILD.SQL.
SQL executed online.
SQL Build log file written to C:\TEMP\PSBLD.log
```

- Log settings:

```
**** Begin Build Settings for this Run
Create Tables = No
Create Indexes = Yes
Create Views = No
Create Triggers = Yes
Alter Tables = Yes
Log to File = Yes
Log to Window = Yes
Write comments to script = No
Always overwrite files = Yes
Execute SQL Now = No
Write SQL to script file = Yes
Logging level = Log fatal errors, warnings and informational messages
Table creation option = Skip table if it exists
View creation option = Drop and recreate view if it exists
IndexCreationOption = Drop and recreate index always
Alter Adds = Yes
Alter Changes = Yes
Alter Renames = Yes
Alter Deletes = Yes
Alter even if no changes = No
Alter Table Option = Alter by Table Rename
Write script output to: Single file
Log filename = c:\temp\PSBUILD.LOG
Single script filename = c:\temp\PSBUILD.SQL
Alter drop column option = skip record if data present
Alter truncate column option = skip record if data would be truncated
Target database platform = MICROSOFT
Target sizing set = 0
```



## Working With the Output Window

You can navigate from error messages in the PeopleSoft Application Designer output window to the corresponding record definition by selecting the appropriate line in the Build log and double-clicking anywhere on that line. PeopleSoft Application Designer opens the corresponding record in the definition workspace.

Also, you can double-click any line in the output window that corresponds to a file, not just a record. PeopleSoft Application Designer opens that file with whatever application that you defined to open files with a particular extension. For example, if you associated SQL files with your native SQL command utility, then your SQL utility opens and loads your PSBUILD.SQL script when you double-click the line that reads:

```
Single script file name = C:\TEMP\PSBUILD.SQL
```

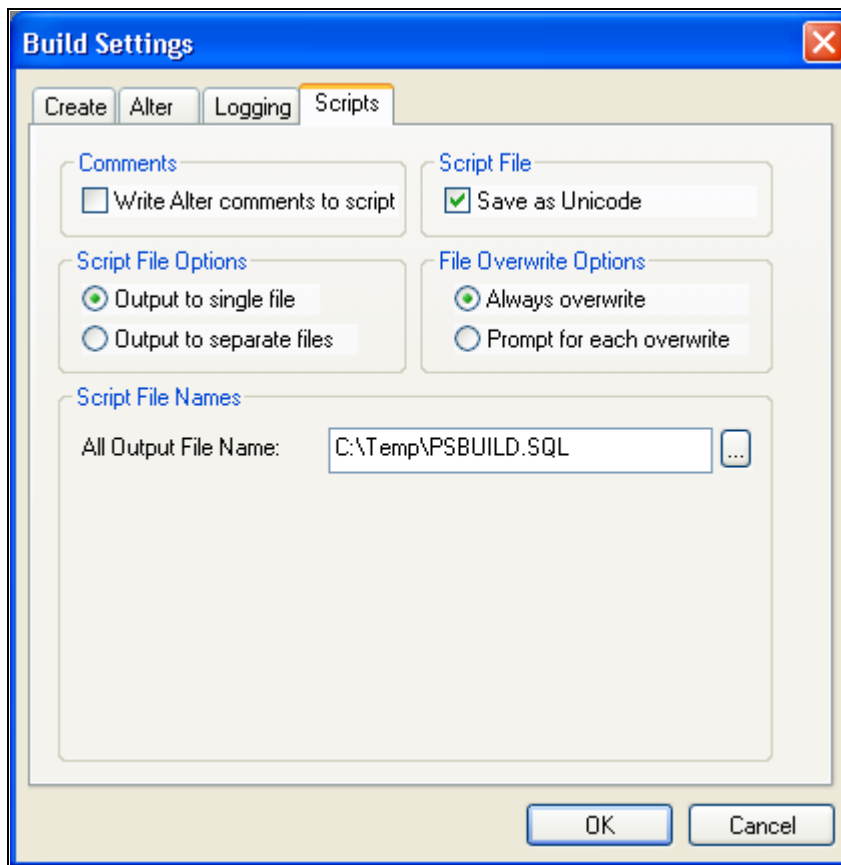
---

**Note.** PeopleSoft Application Designer uses the standard Microsoft Windows method for defining which programs are associated with particular file types. To modify the file associations, open My Computer, select Tools, Folder Options, and then select the File Types tab.

---

## Specifying Script Options

Access the Build Settings dialog box and select the Scripts tab.



Build Settings dialog box: Scripts tab

## Comments

The Write Alter comments to script check box enables you to either include or suppress alter comments. The following example displays the types of comments that you see in your PSALTER.SQL script if you enable this option:

```
-- Alters for record PS_AE_RUN_CONTROL ;
--           AE_THREAD - change ;
--           OPRID - change ;

-- Start the Transaction ;

-- Data Conversion ;

-- Drop Old Indexes ;

DROP INDEX SYSADM.PS_AE_RUN_CONTROL
/

-- Add Columns ;

ALTER TABLE PS_AE_RUN_CONTROL ADD Z_TMP_ALTER_1 DECIMAL(12,4)
/
ALTER TABLE PS_AE_RUN_CONTROL ADD Z_TMP_ALTER_2 CHAR(12)
/
UPDATE PS_AE_RUN_CONTROL SET
    Z_TMP_ALTER_1 = AE_THREAD,
    Z_TMP_ALTER_2 = OPRID
/
```

If you do not view the alter comments, the script containing the same commands as the previous script looks like this:

```
DROP INDEX SYSADM.PS_AE_RUN_CONTROL
/
ALTER TABLE PS_AE_RUN_CONTROL ADD Z_TMP_ALTER_1 DECIMAL(12,4)
/
ALTER TABLE PS_AE_RUN_CONTROL ADD Z_TMP_ALTER_2 CHAR(12)
/
UPDATE PS_AE_RUN_CONTROL SET
    Z_TMP_ALTER_1 = AE_THREAD,
    Z_TMP_ALTER_2 = OPRID
/
```

## Script File

The default for the Save as Unicode check box is based on the UNICODE/ANSI database. This check box is available only when using MSSQLServer, Oracle, DB2/UDB for Linux, UNIX, and Windows, Sybase and DB2 UDB for OS/390 and z/OS databases. Only on Informix , the check box is permanently unavailable because the script files are always ANSI.

The following table lists the default settings for the Save as Unicode check box:

<b>Database Type</b>	<b>Default for Save as Unicode Check Box</b>
Unicode	Selected
ANSI	Unavailable

The script file option, Save as Unicode, is stored in the registry with the other build settings. Therefore, the previous setting is always the default every time that the dialog box is opened.

The type of script file that is generated during the Build process is based on the Save as Unicode check box setting for the UNICODE/ANSI build and UNICODE/ANSI database. This table lists details:

<b>Database Platform</b>	<b>Unicode</b>	<b>ANSI</b>
MSSQL Server	UCS2	ANSI
Sybase	UTF8	ANSI
Oracle	UTF8	ANSI
DB2/UDB for Linux, UNIX, and Windows	UTF8	ANSI
DB2 UDB for OS/390 and z/OS	UTF8	ANSI

### **Script File Options**

If you want all of your CREATE TABLE statements to be written to one SQL script file and your CREATE INDEX statements to another, then select the Output to separate files option. If you prefer to have all of your statements, CREATE, ALTER, and so on, contained in a single file, then select the Output to single file option.

---

**Note.** Script files can be quite large. To reduce the size of files so that they are easier to manage, select the Output to separate files option. Also, the system never appends script files; it overwrites them or you can terminate the process if you elected to be prompted before a build script file is overwritten.

---

Depending on which output option you select, the options in the Script File Names group box change to suit one file or multiple files.

### **File Overwrite Options**

These options indicate whether you want to overwrite automatically or be prompted before script files are overwritten. Your personal preference determines which technique you employ.

**Always overwrite**                      Select to overwrite your previous build scripts.

**Prompt for each overwrite**

Select to be alerted when a script is about to be overwritten.

To avoid the possibility of overwriting files altogether, you can use a naming convention that, in itself, prevents any overwrites. For example, you could specify a unique name for each build script, as shown:

- C:\TEMP\PSINDEX1.SQL
- C:\TEMP\PSINDEX2.SQL

**Script File Names**

The options available to select in this group box depend on your selection in the Script File Options group box.

**Output to single file**

When you select this option, an edit box appears in the Script File Names group box: All Output File Name. The default name for the generated script is PSBUILD.SQL.

---

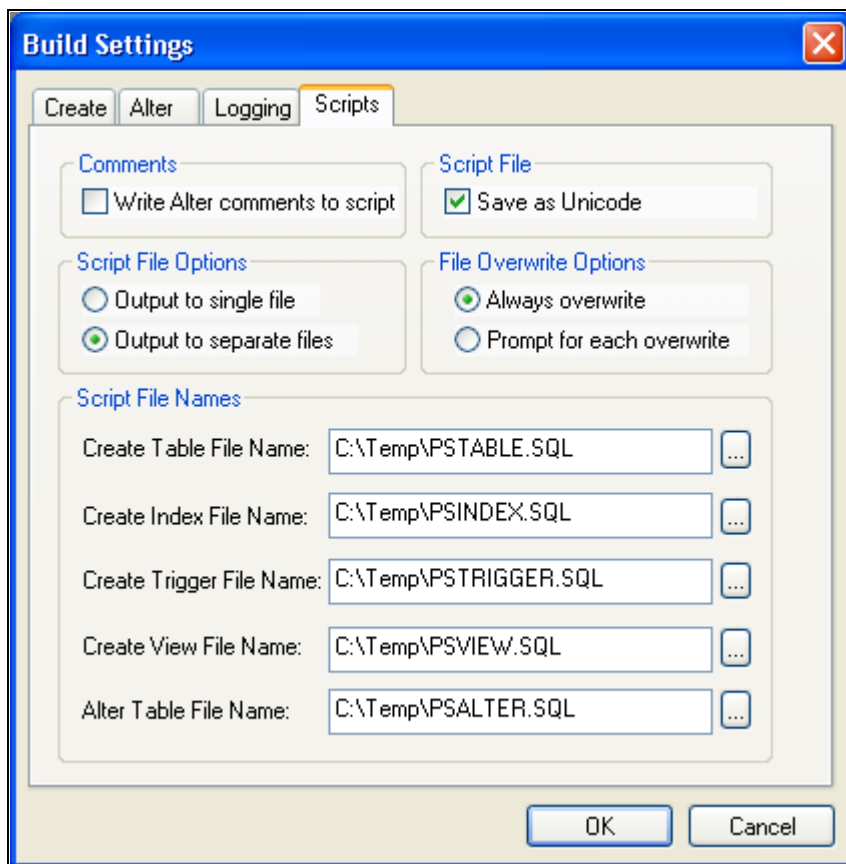
**Note.** Script files can be quite large. To reduce the size of files so that they are easier to manage, select the Output to separate files option. Also, the system never appends script files; it overwrites them or you can terminate the process if you elected to be prompted before a build script file is overwritten.

---

**Output to separate files**

If you select this option, the following edit boxes appear, one for each build option:

- Create Table File Name: The default name for this script is PSTABLE.SQL.
- Create Index File Name: The default name for this script is PSINDEX.SQL.
- Create Trigger File Name: The default name for this script is PTRIGGER.SQL.
- Create View File Name: The default name for this script is PSVIEW.SQL.
- Alter Table File Name: The default name for this script is PSALTER.SQL.



Output to separate files

## Confirming a Table Build

If you are running SQL online, complete Steps 1 and 3; Step 2 is not required.

To confirm a table build:

1. Review the SQL script that was generated by the Build process.

Use your native SQL command processor to open the SQL script. The script is located where you specified on the Scripts tab of the Build Settings dialog box.

2. Run the script against your database.
3. Confirm that the tables now exist.

Use your query tool and SQL Select statements to confirm that the Create Table process created an application table that corresponds to your record definition and updated the system catalog tables. The SQL statement that you use to confirm depends on the table that you just created.

---

## Creating Indexes

This section provides an overview of indexes and discusses how to:

- Configure indexes.
- Add a custom index.
- Edit the DDL.

## Understanding Indexes

Indexes are an important part of your physical database because they affect the efficiency and speed with which your application can store and retrieve data. PeopleSoft application indexes are defined in two ways.

Some indexes are defined for you automatically, based on the key, search key, list box items, and alternate search keys that you specified in your record definition. The application processor uses these indexes for quick access when you use the online system.

However, sometimes you need to define additional indexes to improve the performance of demanding queries, reports, or batch programs. You define and modify these additional indexes using the data administration tools in PeopleSoft Application Designer.

---

**Note.** You can only create indexes for record definition types that are SQL or temporary tables. You would not create indexes on views, such as a SQL view, because views are never physically created. The Indexes option in the Tools, Data Administration menu is disabled when a view is open.

---

When you use SQL Create or SQL Alter for a table, the system automatically builds database indexes to the SQL table that are based on the keys and list items that you specify in the record definition. Your database uses indexes to find definitions in the database the same way that you use indexes to find specific information in a book. If an index entry exists for a specific field in a table, the database can search and quickly find what it needs to complete your request; otherwise, it must search the contents of the entire table.

Indexes enhance system performance by generating key lists that the application processor uses to extract the data that it uses to populate pages and search records. The system automatically creates:

- Standard indexes (key or duplicate order key indexes) if at least one field in the table is a key or duplicate order key.

The index contains all key and duplicate order key fields. The system automatically names this index with a prefix of *PS\_*.

- Alternate search key indexes for each alternate search key.

The system automatically names this index with a prefix of *PSn*, in which *n* is a number between 0 and 9.

- Custom indexes, which you can define using the Edit Index, Edit DLL, and Add Index dialog box in the PeopleSoft Application Designer administration tools.

The system automatically names this index with a prefix of *Psa*, in which *a* can be any letter between A and V. The system does not create custom indexes automatically.

For most database tables (those with unique keys), the key index defines unique keys for a table, but the system also uses it to access high-level keys and orders by all keys. Alternate search key indexes support the search record and field prompts. The purpose of these different indexes is to enable the system to respond to all system prompts through index-only SQL access, minimizing direct database table access.

---

**Note.** If you use Create Table or Alter, the Create Index option is automatically selected. With the Create Table or Alter options, you cannot deselect the Create Index option.

The alter process takes care of indexes regardless of the Alter Table Options you select on the Alter tab of the Build Settings dialog box. In the case of the Alter by Table Rename option, the indexes are dropped automatically when the temporary table is dropped. For the Alter in Place option, the index creation process goes through the Recreate index only if modified option on the Create tab.

When creating indexes with the Alter method, before altering the table, drop the statistics using "DROP STATISTICS" that were created explicitly in columns using "CREATE STATISTICS."

---

The following record definition modifications require you to create new indexes:

- Add, delete, or modify keys, duplicate order keys, alternate search keys, or descending keys in the Record Field Properties dialog box.
- Change the order of keys (ascending, as opposed to descending) in the Change Record Indexes dialog box.
- Change the Unique settings in the Edit Index dialog box.

### ***Platform Support for Indexes***

Most database platforms support indexes with an unlimited number of keys. However, Microsoft SQL Server 2000; DB2/UDB for Linux, UNIX, and Windows (DB2/UDB LUW); and Informix support only indexes that have a maximum of 16 keys. PeopleSoft Application Designer supports indexes for these databases by creating functional indexes. If a record definition has more than 16 keys, the Build process automatically alters the table to support functional indexes. The system adds a warning message to the log file indicating this. You can find sample code for functional indexes for each of these platforms in Appendix A of this PeopleBook.

See [Appendix A, "Understanding Functional Indexes," page 573](#).

The following list offers more details about functional indexes by database platform:

- Functional indexes for Microsoft SQL Server 2000.

If an index has more than 16 keys, the key fields are concatenated to a functional key. This key field is then added to the table as the last column, called MSSCONCATCOL. A unique index is created for MSSXCONCATCOL as well as an additional index called PSW<record name>, which is used as a search index.

- Functional indexes for DB2/UDB LUW.

Similar to the concatenation that occurs with SQL Server, DB2/UDB LUW also concatenates the key fields to a functional key. The key is then added to the table as the last column, called DBXCONCATCOL. A unique index is created for DBXCONCATCOL as well as an additional index called PSW<record name>, which is used as a search index.

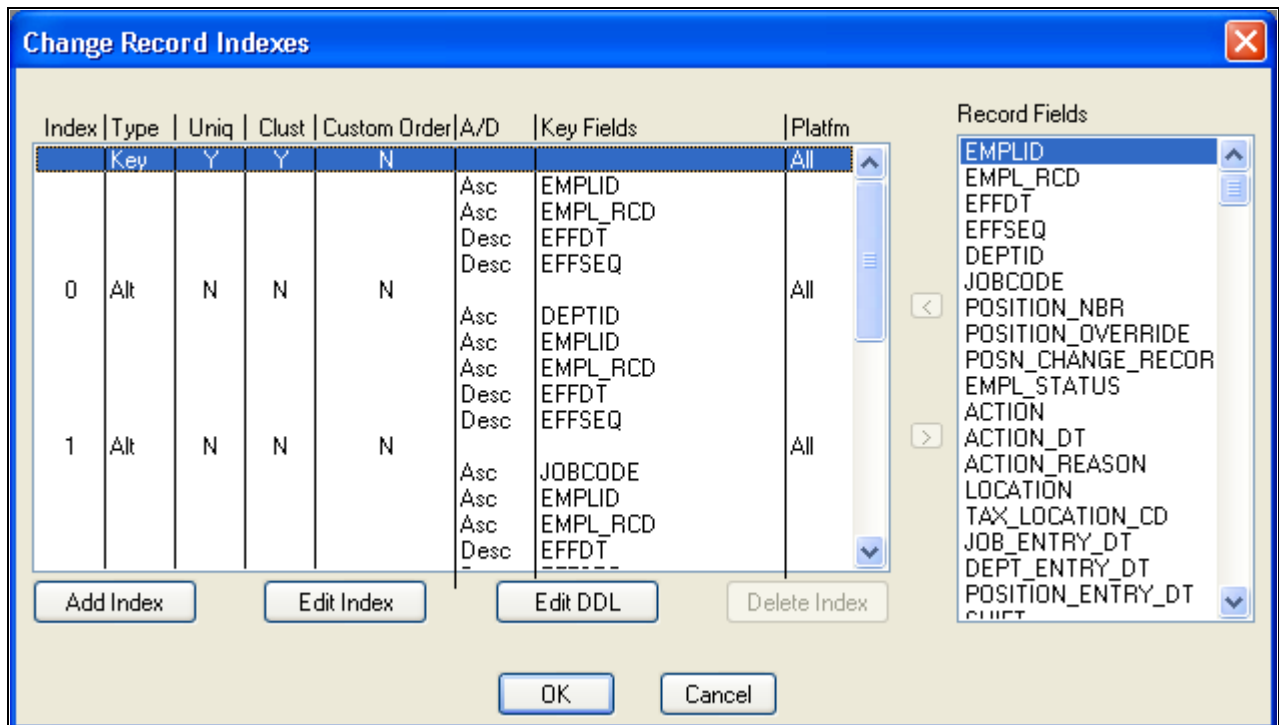
- Functional indexes for Informix.

If an index has more than 16 keys, a function is created for every 16 keys. The name for this set of 16 keys is PS\_<record name>\_SP<the incremental number for each function>. The functions are used to create a unique index. An additional index, called PSW<record name>, is also created.

## Configuring Indexes

For performance tuning, you might want to change the order of your index keys. PeopleSoft Application Designer provides direct control over all of the indexes created by PeopleTools. You can inspect, add, or change all defined indexes or delete custom indexes for any record in the system through the Change Record Indexes dialog box.

Standard indexes are based on the key, duplicate order key, or alternate search key only. However, with the Change Record Indexes dialog box, you can modify those indexes and create additional indexes, which might be required if you do custom processing. Any of the indexes that you entered can be maintained for all database platforms or for a list of specific platforms. All indexes and keys created by PeopleTools are stored in PeopleSoft system tables.



Change Record Indexes dialog box

To edit the index key order:

1. Open the record definition for which you want to make the custom index.
2. Select Tools, Data Administration, Indexes to access the Change Record Indexes dialog box.

The Custom Order column indicates whether the index keys were configured. The default selection is *N*, indicating that no index keys were configured.

3. Double-click *N* or *Y* in the Custom Order column for the index that you want to alter, or click Edit Index.
4. Select the Custom Key Order check box.
5. Click OK.

The Custom Order entry changes from *N* to *Y*.



6. Drag and drop the field that you want to move into the appropriate order.

After configuring the key order, to reset the key order of the index to the original order as defined in the record definition, deselect the Custom Order entry (Y to N), and click OK. The order of the key fields reset to the original position in the Change Record Indexes dialog box.

7. Check the key order in the Use Display mode of your record definition.

The Num (number) column represents the original order of the keys in your record, and the Ord (order) column reflects the key order in the index.

8. Create the indexes:

- a. Select Build, Current Project.
- b. Select Create Indexes in the Build Options group box.

Make sure that you select the appropriate build settings.

9. Run the Build process to either generate a script or run the SQL online.
10. Confirm the construction of the new index with your native SQL command processor.

---

**Note.** Users might also configure indexes for parent records that contain one or more subrecords without key fields, such as the ADDRESS\_SBR subrecord. However, if the subrecords contain key fields, then index key order changes are restricted. The Custom Key Order option is deactivated and cannot be changed from N to Y. Change the index key order for the parent before inserting the subrecord.

---

## Adding a Custom Index

To add a custom index:

1. Open a record.
2. Select Tools, Data Administration, Indexes to access the Change Record Indexes dialog box.
3. Click Add Index.

The Add Index dialog box enables you to determine for which database platforms you want to define the index if you selected the Some option for Platform. The system selects the Custom Key Order check box to ensure that the underlying record definition is not adversely affected by the change to your index.

For all user-defined indexes, you must enter a description of the index in the Comments section. You should know the purpose of the index, the transactions and processes that use this index, and other relevant information. This information assists you when you want to remove unused indexes. An error message appears if you do not remove unused indexes.

---

**Note.** You cannot add more than 22 user indexes (A–V). The system disables the Add Index button if the record already contains 22 user indexes.

---

4. Indicate the database platforms for which you want to create an index.
5. Click the OK button to add a row to the Change Record Indexes dialog box and to enable the arrow buttons.

6. Move the field under the new index row:
  - a. In the Record Fields list box, select the field that you want indexed.
  - b. Click the arrow button or double-click the field.
  - c. Repeat this step for each field that you want to add.
7. Click OK.
8. Save the record definition.
9. Select Build, Current Project.
10. Select Create Indexes under Build Options.

Make sure that you select the appropriate build settings.
11. Run the Build process to either generate a script or run SQL online.
12. Run the generated SQL script if you opted to generate one.
13. Confirm the construction of the new index with your native SQL command processor.

## Editing the DDL

In the Change Record Indexes dialog box, you can view the DDL for the index and override DDL parameters that are defined in the DDL model for this index.

To edit the DDL:

1. In the Change Record Indexes dialog box, select the index that you want to edit.
2. Click the Edit DDL button to access the Maintain Index DDL dialog box.

The upper list box in this dialog box lists the platform and DDL parameter, and the lower list box shows the DDL templates for the various platforms. Your database might have fewer platforms enabled. The sizing set enables different collections of tables to have different model statements and parameters.

3. Select the platform and sizing set.
4. Click View DDL.

The dialog box that appears shows the index DDL for the platform and sizing set that you selected. Review this DDL statement.

5. Click the Close button when you are finished.
6. Select the parameter that you want to edit in the upper list box.

The default value is set in the DDL Model Defaults dialog box. You can change this setting by selecting Go, PeopleTools, Utilities, Use, DDL Model Defaults in PeopleSoft Application Designer, or by selecting PeopleTools, Utilities, Use, DDL Model Defaults in the browser.

7. Click the Edit Parm (edit parameter) button to access the Edit Override Parm Value (edit override parameter value) dialog box.

8. Enter the override value for the parameter that you selected.
9. Select OK for both open dialog boxes.

---

## Altering Tables

This chapter provides an overview of the Alter process and discusses how to:

- Determine tables to be altered after a field change.
- Alter a table.
- Specify alter options.
- Confirm your table alter.

## Understanding the Alter Process

The Alter process enables you to make changes to your record definitions at any time without losing the application data that is already housed in the table. To ensure that the Alter process produces the appropriate results, be aware of when to alter an existing table and what types of conversions PeopleSoft Application Designer supports. To preserve data that is stored in tables and also reflect a recent change in the table definition, use the alter tables function to make the appropriate changes.

The alter function simplifies the process of modifying database tables when new data fields are added or when existing data fields are modified. It effectively eliminates the need to write SQL statements to perform the same function.

The alter process was specifically designed to automate the task of writing alter scripts and to protect the integrity of your database. In essence, it protects your interests by ensuring that you always control data loss. The alter function performs tests that verify whether data will be lost during the conversion of each column to be altered. Data loss normally occurs when reducing the character width of a column, dropping a column that contains data, or converting a number field to a character field that is too small.

The generated statements for this method are sent to a SQL script file, the alter script. After the script has completed, you are responsible for reapplying any SQL table DDL that you applied outside of PeopleTools. Typically, this SQL table DDL consists of GRANT and REVOKE statements that control security.

---

**Note.** In Oracle databases, procedural language/SQL (PL/SQL) is required to use the PeopleSoft Application Designer Build process.

---

### ***Alter Settings***

The Alter tab in the Build Settings dialog box handles the most common types of data conversions, including:

- Increasing the length of character fields.
- Changing a character (CHAR) field to a long character (LONG VARCHAR) field.
- Increasing the size of number or character fields.

- Reducing the decimal positions in number fields.

If any column of the table has data that will be lost when truncating, the system action is determined by the Change Column Length options.

- Changing integer-only number fields to character fields, where the character field is formatted with leading zeros.

If any column of the table has a number that does not fit in the new character field, the system action is determined by the Change Column Length options to prevent data from being lost.

- Adding columns to SQL tables.
- Dropping columns from SQL tables.
- Renaming columns in SQL tables.
- Renaming SQL tables.
- Moving tables to a new tablespace.

The following sections describe the types of alter conversions that you can expect to occur during the Build process.

### Data Conversions

When altering tables with existing field data, PeopleSoft Application Designer handles conversions as outlined in the following table. A *No* value indicates that conversion cannot take place. A number in the cell following a *Yes* value indicates that restrictions are involved with the conversion. Refer to the corresponding numbered note following the table for additional information:

PeopleSoft Previous Field Type Format	PeopleSoft New Field Type Format										
—	Char	Oracle - Long Char with Length < 2000	Long Char	Image	Long Char - Raw Binary	Number/Signed Number where n = length and d = decimal			Date	Time	DateTime
						Small Integer * (small integer)	Integer **	Decimal ***			
Char	Yes (1)	Yes (1)	Yes (9)	No	No	No	No	No	No	No	No

<b>PeopleSoft Previous Field Type Format</b>	<b>PeopleSoft New Field Type Format</b>										
<i>Oracle - Long Char with Length &lt; 2000</i>	Yes (1)	Yes (1)	Yes	No	No	No	No	No	No	No	No
<i>LongChar</i>	Yes (2) (9)	Yes (2)	Yes (8)	No	No	No	No	No	No	No	No
<i>Image</i>	No	No	No	Yes (8)	Yes (8)	No	No	No	No	No	No
<i>Long Char - RawBinary</i>	No	No	No	Yes (8)	Yes (8)	No	No	No	No	No	No
<i>SmallInt *</i>	Yes (5)	Yes (5)	No	No	No	No Action	Yes (4)	Yes (4)	No	No	No
<i>Integer **</i>	Yes (5)	Yes (5)	No	No	No	Yes (4)	No Action	Yes (4)	No	No	No
<i>Decimal ***</i>	Yes (7)	Yes (7)	No	No	No	Yes (3)	Yes (3)	Yes (3)	No	No	No
<i>Date</i>	No	No	No	No	No	No	No	No	No Action	No	No
<i>Time</i>	No	No	No	No	No	No	No	No	No	No Action	No
<i>DateTime</i>	No	No	No	No	No	No	No	No	Yes (6)	Yes (6)	No Action
Notes:											

<b>PeopleSoft Previous Field Type Format</b>	<b>PeopleSoft New Field Type Format</b>		
	* n < 4, d = 0, n = 5, d = 0 RawBinary	** n > 4, d = 0, (or) n = 10, d = 0 RawBinary	*** n > 9, d > 0
1	If data fits or data truncation is allowed.		
2	Not allowed in Oracle.		
3	If data fits or data truncation is allowed (for decimal portion only).		
4	If data fits. No data truncation can occur, even with the option to allow data truncation.		
5	If data fits, except for Informix.		
6	If data truncation is allowed by selecting the Truncate data if field too short option on the Alter tab of the Build Settings dialog box.		
7	If data fits, except if data has decimal values or is for Informix.		
8	If image or long character change from Null to Not Null or Not Null to Null, otherwise, no action.		
9	Allowed on all platforms except Informix (due to an Informix limitation).		

### **Alter Process Tips**

As you create and run alter scripts, keep the following information in mind.

- Temporary tables during alter:

When you run the Alter process, the script might create a temporary table. Temporary tables are named PSY%tablename%, and they are created in the tablespace of the altered table, that is, the tablespace currently defined in the record definition.

- Alter process on tables containing LONGS on Oracle:

Oracle does not allow an INSERT or SELECT FROM command if a table contains a LONG VARCHAR or LONG RAW data type. Therefore, in Oracle databases, PeopleSoft Application Designer uses an Oracle PL/SQL script to perform alters on tables that contain any type of LONG data type.

- Alter process on Oracle on records with CHAR fields greater than 1333 bytes but less than 2000 bytes:

For Oracle, records with CHAR fields of length greater than 1333 characters and less than 2000 characters will appear in the alter SQL script as needing to be altered, even if the information has not changed. In Oracle Unicode databases, PeopleSoft applications triple the length of a field while creating a table with a VARCHAR2 column in the database to store all three bytes of the Unicode character. In Oracle databases, VARCHAR2 can store up to 4000 bytes, which means a restriction exists to store only 1333 characters (with three bytes each).

For example, if a LONG CHAR field is designed with a length of 1999 characters, then PeopleSoft applications interpret the field as a VARCHAR2 and create the column with a size of 1333 characters. The field length of that LONG CHAR field will be stored as 1999 in the PSRECFIELD table, but in the database it will be created with a size of 1333. So, each time a record with a LONG CHAR field of 1999 (or any value > 1333), the alter process detects the difference between the 1999 character length in PSRECFIELD and the 1333 character length in the database and generates the alter script. In addition, PeopleSoft applications also create a constraint on the VARCHAR2.

- View dependencies:

PeopleSoft Application Designer does not keep track of view dependencies. When the structure of a table changes, you should recreate all views. Many database environments track this information, so you might be able to determine view dependencies by issuing a query directly against the system catalog. However, if you are not the DBA, you might not have the authority to run such a query successfully.

Also, renaming or dropping a table, view, or column might invalidate view text, dynamic view text, and possibly other stored queries.

This table describes what happens to views by database platform:

<b><i>Database Platform</i></b>	<b><i>What Happens to the Dependent Views When a Table Is Dropped?</i></b>	<b><i>Do the Inoperative Views Become Operative When the Table Is Recreated?</i></b>	<b><i>Comments</i></b>
Microsoft SQL Server	Becomes inoperative.	Yes	The dependency reported by sp_depends is incorrect until you recreate the view.
Sybase	Becomes inoperative.	Yes	The dependency reported by sp_depends is incorrect until you recreate the view.
Oracle	Becomes inoperative.	Yes, but only after the first use of the dependent view.	The VIEW object that became invalid remains invalid until its first use. The Oracle catalog table for objects marks the view as valid again.
Informix	Is dropped when the table is dropped.	Not applicable (NA).	Views need to be created because the system drops them along with the table drop.
DB2/UDB for Linux, UNIX, and Windows	Becomes inoperative.	No	Views need to be recreated to make them operative.
DB2 UDB for OS/390 and z/OS	Is dropped when the table is dropped.	NA	Views need to be recreated because the system drops them along with the table drop.

- Alter script:

After you invoke the Alter process, the system uses the default value that is defined in the record definition to populate the column in the altered table. The alter script that is generated contains detailed comments indicated by the (--) to assist you if you must modify the script manually.



- Preserve storage parameters on Oracle databases:

Alter by table rename preserves customized storage parameters for tables only (with some limitations) not for indexes, by taking DDL parameters from the system catalog.

### ***When to Alter Tables***

The following record definition changes affect synchronization with the application table and require an alter process:

- Add or delete a field in the record.
- Modify the length of a field.
- Change the required status of a date, time, DateTime, or long field.
- Rename a field.
- Rename a record.

The Alter Table process is similar to the Create Table process except that it does not drop existing application data tables and the data that they contain.

---

**Note.** The Build process automatically alters records with more than 16 keys to support functional indexes. A warning message indicating this is included in the log file for Microsoft SQL Server and DB2/UDB for Linux, UNIX, and Windows. For Informix, the warning message reads: "The record %s has more than 16 keys. Indexes will be re-created to support functional indexes."

---

### ***Working With Alters on DB2 for z/OS***

You must be aware of a variety of considerations and maintenance tasks when performing alters on DB2 UDB for z/OS. For example, you must consider:

- Tablespace versioning.
- Advisory Reorg Pending status.
- Rebuild Pending status.
- Avoiding SQL Code -4702.

These topics are discussed in detail in the *PeopleTools 8.52 PeopleBook: Data Management*.

See *PeopleTools 8.52: Data Management*, "Administering PeopleSoft Databases on DB2 UDB for z/OS," Working with Alters on DB2 z/OS.

## **Determining Tables To Be Altered After a Field Change**

To determine tables to be altered after a field change:

1. Create a new project.
2. Open the altered field.

3. Select Edit, Find Definition References.

The referenced definitions appear in the output window.

4. Select all of the rows that appear in the output window.
5. Right-click the selected rows and select Insert Into Project from the pop-up menu.

## Altering a Table

To alter a table:

1. Open the project for which you must perform an alter.
2. From the Build menu, select the appropriate scope of your build.

The Build dialog box appears.

3. Select the Alter Tables option in the Build Options group box.

Selecting the Alter Tables option automatically selects and disables the Create Indexes and Create Trigger (only if triggers are needed) option. In the Build Execute Options group box, only the Build script file option is enabled.

4. Click the Settings button in the Build dialog box to set user-defined defaults.

Select the appropriate alter, logging, and scripts settings.

---

**Note.** The alter process takes care of the indexes, regardless of the Alter Table Options you select on the Alter tab of the Build Settings dialog box. In the case of the Alter by Table Rename option, the indexes are dropped automatically when the temporary table is dropped. For the Alter in Place option, the index creation process goes through the Recreate index only if modified option on the Create tab.

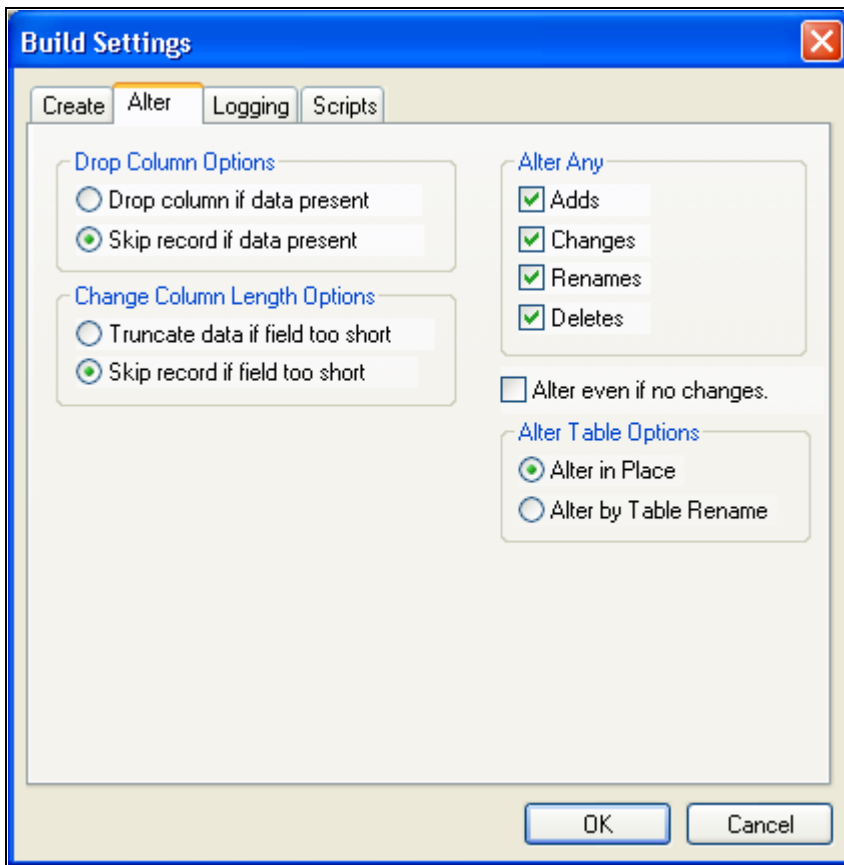
---

5. In the Build dialog box, click Build to run the Alter Tables process.

The length of a build process depends on the number of definitions that require building. Watch the Build Progress dialog box to monitor the Build process. When the process completes, check any errors listed in the log file.

## Specifying Alter Options

Access the Build Settings dialog box and select the Alter tab.



Build Settings dialog box: Alter tab

### ***Drop Column Options***

These options are referenced whenever a field is deleted from a PeopleTools record definition in which data exists in the database for that column.

**Drop column if data present**     Select to drop the column and data, and write a warning to the build log.

**Skip record if data present**     Select to end the alter for that record and write an error message to the log. Processing continues with the next record.

---

**Note.** Whenever you select the Drop column if data present option, you run the risk of losing data, and you are prompted at runtime to confirm your selection of this option.

---

### ***Change Column Length Options***

Use these options whenever the length of a character column is reduced in PeopleTools and an alter could result in data truncation.

**Truncate data if field too short**     Select to alter the record and write a warning message to the build log. Note that the system does not truncate data for numeric fields because of possible data loss.

**Skip record if field too short** End the alter for that record and write an error message to the build log. Processing continues with the next record.

---

**Note.** Whenever you select the Truncate data if field too short option, you run the risk of losing data, and you are prompted at runtime to confirm your selection of this option. Also, numeric fields are not truncated to avoid any loss of data.

---

### ***Alter Any***

PeopleTools assumes that you want to perform alters for any modifications made to tables, so, by default, all of the check boxes in the Alter Any group box are selected. The Alter Any group box enables you to customize alter processing for adds, changes, renames, and deletes. For example, you have the flexibility to add, change, and rename fields but not to delete any removed columns. When you complete any other custom external conversion processes, you can then enable the delete processing to drop columns that are no longer defined.

Normally, you select all of these options, but during upgrades or operations requiring data conversion steps you might select a subset of actions. For example:

- Perform alter with Adds and Changes selected.
- Perform data conversion routines to populate the new and changed columns (perhaps from columns that are ultimately to be dropped).
- Perform alter with Rename and Delete selected.

### ***Alter even if no changes***

Select this check box to force an alter, even if no changes were made to the tables. This check box is deselected by default. If selected, the Alter by Table Rename option in the Alter Table Options group box is automatically selected and the Alter in Place option is disabled.

### ***Alter Table Options***

When altering tables, the alter process takes care of the indexes, regardless of the Alter Table Options you select on the Alter tab of the Build Settings dialog box. In the case of Alter by Table Rename, the indexes are dropped automatically when the temporary table is dropped. For the Alter in Place option, the index creation process goes through the Recreated index only if modified option on the Create tab.

- Alter in Place

Selected by default. For database platforms in which Alter in Place is not supported, alter is automatically done by Alter by Table Rename (even if that option is not selected).

Database platforms vary in the capabilities for Alter in Place table operations. This table displays the limitations for each operation by platform:

<b>Database Platform</b>	<b>Add Column</b>	<b>Change Column (data type, size, precision, scale, and null/not null property change)</b>	<b>Rename Table</b>	<b>Delete Column</b>
Microsoft SQL Server	Yes *	Yes *	Yes *	Yes *
Oracle	Yes *	No	Yes	Yes *
DB2/UDB for OS/390 and z/OS	Yes *	Yes *	Yes *	No
DB2/UDB for Linux, UNIX, and Windows	Yes *	No	No	No
Sybase	No	No	Yes	No
Informix	Yes	Yes	Yes	Yes
* Certain restrictions might apply to this operation. Refer to the documentation for your specific database platform for more information.				

- Alter by Table Rename

Select to create a temporary table (with changes made to the original table or its fields); the data from the original table is imported into the temporary table. The original table is then dropped and the temporary table is renamed to the original table.

---

**Note.** For Oracle databases only, Alter by Table Rename preserves customized storage parameters for tables (with some limitations), by taking DDL parameters from the system catalog.

---

If a table is renamed from the old name to a new name, the indexes that were created for the old table are moved to the new table, but the index names remain in the name of the old table. With Alter by Table Rename selected, the indexes of the old table are dropped before being renamed to the new table name, and the indexes are recreated after the table is renamed to the new name. This way, the index is created in the name of the new table.

See *PeopleTools 8.52: Data Management*, "Administering PeopleSoft Databases on DB2 UDB for z/OS," Working with Alters on DB2 z/OS.

## Confirming Your Table Alter

To confirm your table alter:

1. Review the SQL script that was generated by the Alter process.  
  
Use your native SQL command processor to open the SQL script. The script is located where you specified on the Scripts tab of the Build Settings dialog box.
2. Run the script against your database.
3. Use your query tool and SQL Select statements to confirm that the Alter Tables process created an application table that corresponds to your record definition and updated the system catalog tables.

---

## Creating Triggers

A database trigger is a type of stored procedure that runs automatically when a user makes a specific data-modification statement (INSERT, UPDATE, or DELETE) on a specified table. Within PeopleTools, you create triggers in tables related to synchronization for mobile applications and for optimization servers. Do not create triggers for all records; create them only for records that have specific properties set.

For example, in a mobile component record, set the required properties for that record so that triggers are created during the Build process. Using the Build dialog box, create the trigger by selecting the Create Trigger check box in Build Options. Using the Build Execute options, you can either generate the SQL or run the build online. Usage and implementation restrictions exist.

---

**Important!** PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.

---

### See Also

*PeopleTools 8.52: PeopleSoft Optimization Framework*, "Designing Analytic Type Definitions," Creating and Building Optimization Records

---

## Creating Views

This section provides an overview of online views and discusses how to:

- Create views.
- Use joins in views.
- Use cross-platform meta-SQL for dynamic views.

## Understanding Online Views

Views in PeopleSoft applications are comparable to views used in traditional relational database management system (RDBMS) technology. A view, in essence, is a virtual table. It is created from the results of a query run against one or more tables.

When you create views to use online, you must keep the views synchronized with the database. Like SQL tables, you must have a record definition for online views because the system checks the record definitions for online processing rules. You must build the view before you can use or reference it online.

If you use a view as the basis for a page, you select existing information to display on the page. Therefore, when you create the record definition for your view, you can clone an existing record definition, delete any fields that are not used in the view, and proceed to define the view. Query views are processed the same way as view text, which is defined as part of the record.

The order in which PeopleTools creates views is based on the build sequence number that you set on the Record Type tab while the record is open. The default build sequence number is 1. For the dependent views to be created first, the build sequence number should be 0. This way, all of the 0 views are created first, then the 1 views, and finally the views that are greater than 1. The maximum build sequence that can be set is 99. Although the views are sorted alphabetically in the project, they are created based on the build sequence number.

### See Also

Chapter 6, "Creating Record Definitions," Creating SQL View and Dynamic View Select Statements, page 152

Chapter 6, "Creating Record Definitions," Understanding Record Types, page 117

## Creating Views

If you are already familiar with creating record definitions and writing SQL queries, creating online views is similar.

To create online views:

1. Clone a record definition:
  - a. Open an existing record definition.
  - b. Select File, Save As.
  - c. Specify the name of the view that you want to create.
2. Delete and add the appropriate fields.

All columns that are selected in your view should have a corresponding field in the record definition.

3. Set the record type to SQL View:
  - a. Select the Record Type tab for the record definition.
  - b. Select SQL View in the Record Type group box.
4. Click the Click to open SQL Editor button.

The order in which you list the columns in your SQL SELECT statement should match the order that you specified in the record definition. A SQL statement can be given for each platform.

See [Chapter 6, "Creating Record Definitions," Creating SQL View and Dynamic View Select Statements, page 152.](#)

5. Save the record definition.
6. Select Build, Current Definition to create the view in the database.
7. Select the Create Views check box under Build Options.
8. Click Build.

## Using Joins in Views

If you plan to use a join regularly, you can save it as a SQL script or you can create a view—a single virtual table—using the join as the logical representation of the data. After you create a view, the users never need to know that the data they are viewing is stored in multiple tables. To users, the relative complexity of your view is transparent. For example:

```
SELECT  A.EMPLID,
        A.NAME,
        B.CONTACT_NAME,
        B.RELATIONSHIP
FROM    PS_PERSONAL_DATA A,
        PS_EMERGENCY_CNTCT B
WHERE   A.EMPLID = B.EMPLID;
```

## Using Cross-Platform Meta-SQL for Dynamic Views

PeopleTools includes functionality to produce meta-SQL constructs for dynamic views, which means that dynamic SQL views that are created using PeopleSoft Application Designer can be used on any platform that is supported by PeopleSoft. The *PeopleSoft PeopleTools PeopleBook: PeopleCode Developer's Guide* contains a full list of the meta-SQL that is supported in the PeopleSoft Internet Architecture.

### See Also

*PeopleTools 8.52: PeopleCode Language Reference*, "Meta-SQL Elements," Meta-SQL Placement Considerations



## Administering DDL

Select Data Administration from the Tools menu in PeopleSoft Application Designer to access critical dialog boxes that enable you to define the record location and structure and other guidelines for PeopleTools to extract information from your selected database.

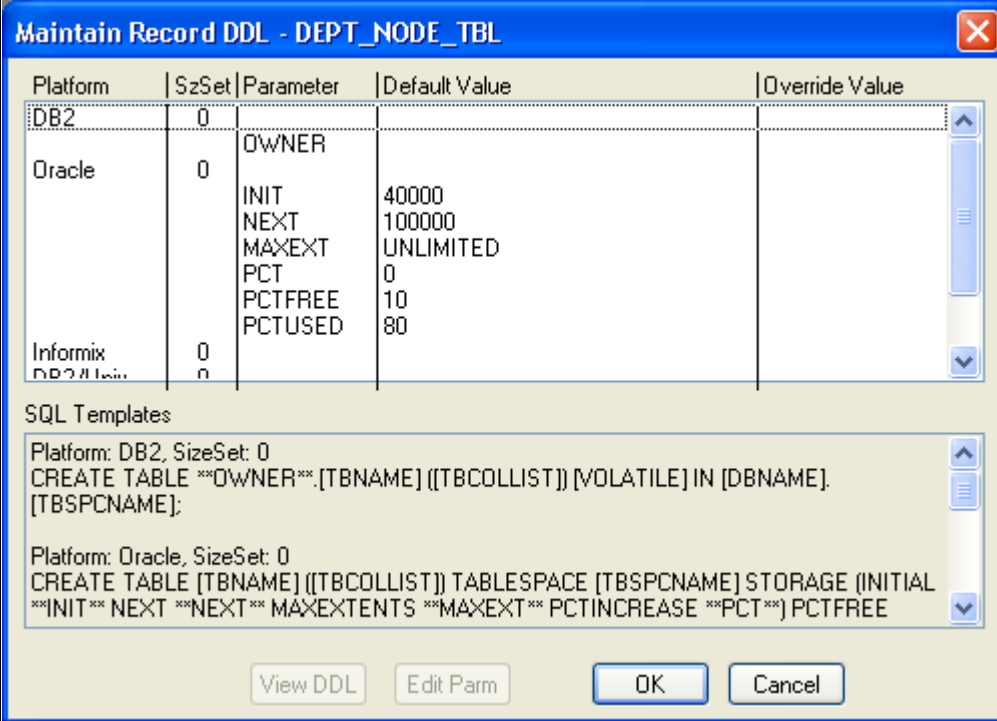
This section discusses how to:

- Use the record DDL.
- Set the tablespace.
- Use physical data storage.

## Using the Record DDL

Use the record DDL to define parameters and default values for the tables in your database. PeopleSoft applications provide templates for each database platform that PeopleTools supports. The templates contain the typical parameters for each database platform. The mechanics of editing and viewing record, index, and unique index DDL are the same as for space DDL.

Access the Maintain Record DDL dialog box from Tools, Data Administration, Record DDL. The Data Administration menu option is only enabled if a record definition is open.



The dialog box titled "Maintain Record DDL - DEPT\_NODE\_TBL" displays a table of parameters for different database platforms. The table has five columns: Platform, SzSet, Parameter, Default Value, and Override Value. The rows show parameters for DB2, Oracle, and Informix. Below the table, there is a section for SQL Templates with two templates: one for DB2 and one for Oracle. At the bottom, there are buttons for "View DDL", "Edit Parm", "OK", and "Cancel".

Platform	SzSet	Parameter	Default Value	Override Value
DB2	0	OWNER		
Oracle	0	INIT	40000	
		NEXT	100000	
		MAXEXT	UNLIMITED	
		PCT	0	
		PCTFREE	10	
		PCTUSED	80	
Informix	0			

SQL Templates

Platform: DB2, SizeSet: 0  
 CREATE TABLE \*\*OWNER\*\*.[TBNAME] ([TBCOLLIST]) [VOLATILE] IN [DBNAME].  
 [TBSPCNAME];

Platform: Oracle, SizeSet: 0  
 CREATE TABLE [TBNAME] ([TBCOLLIST]) TABLESPACE [TBSPCNAME] STORAGE (INITIAL  
 \*\*INIT\*\* NEXT \*\*NEXT\*\* MAXEXTENTS \*\*MAXEXT\*\* PCTINCREASE \*\*PCT\*\* PCTFREE

View DDL Edit Parm OK Cancel

Maintain Record DDL dialog box

**View DDL**

Select a Platform row in the Maintain Record DDL dialog box to enable the View DDL button. This button opens the DDL Statements dialog box, which displays the CREATE TABLE DDL for the current record on each of the database platforms.

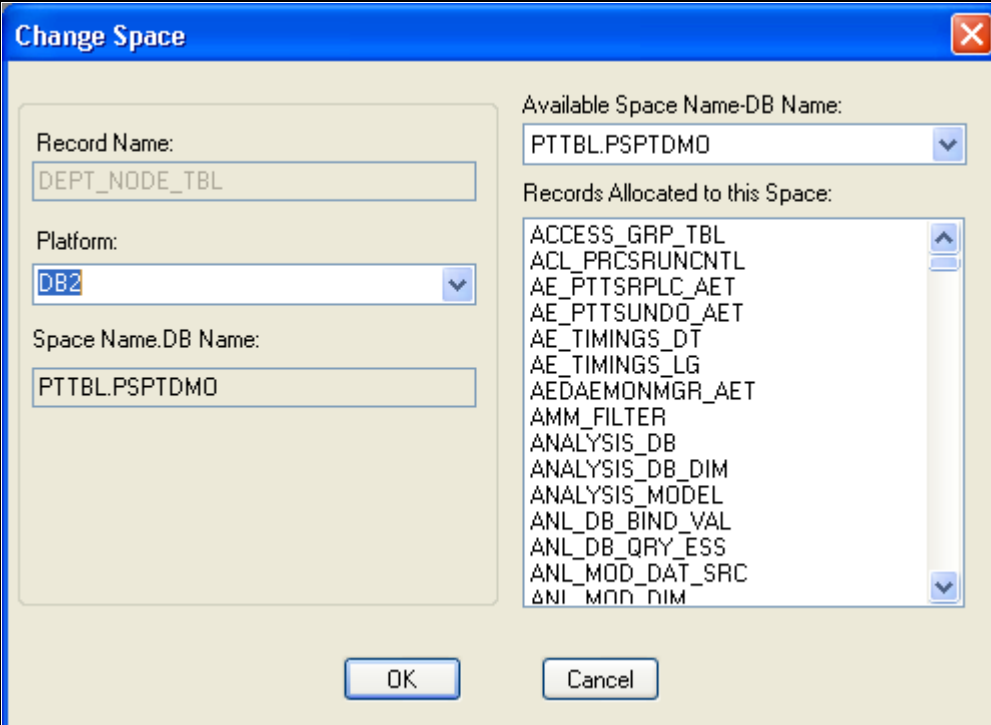
**Edit Parm(edit parameter)**

Select a Parameter row in the Maintain Record DDL dialog box to enable the Edit Parm button. This button opens the Edit Override Parm Value dialog box, where you can enter a new value to override the Default Value listed in the table. The new value is stored in the PSRECDDLPRM override table.

## Setting the Tablespace

The Change Space dialog box is an editing tool with which you can select the space name and view the records attributed to the named space in the database.

Access this dialog box by selecting Tools, Data Administration, Set Tablespace. This dialog box also appears automatically when you attempt to save a new record definition. For each new record definition that is created in PeopleSoft Application Designer, you must allocate a tablespace name. For the Change Space dialog box to appear upon saving a new record definition, you must select the Platform Compatibility Mode check box under PeopleTools, Utilities, Administration, PeopleTools Options.



The screenshot shows the 'Change Space' dialog box. It has a title bar with a close button. The dialog is divided into two main sections. The left section contains three input fields: 'Record Name:' with the text 'DEPT\_NODE\_TBL', 'Platform:' with a dropdown menu showing 'DB2', and 'Space Name.DB Name:' with the text 'PTTBL.PSPTDMO'. The right section contains a dropdown menu labeled 'Available Space Name-DB Name:' with 'PTTBL.PSPTDMO' selected, and a list box labeled 'Records Allocated to this Space:' containing a scrollable list of table names: ACCESS\_GRP\_TBL, ACL\_PRCRUNCTL, AE\_PTTSRPLC\_AET, AE\_PTTSUNDO\_AET, AE\_TIMINGS\_DT, AE\_TIMINGS\_LG, AEDAEMONMGR\_AET, AMM\_FILTER, ANALYSIS\_DB, ANALYSIS\_DB\_DIM, ANALYSIS\_MODEL, ANL\_DB\_BIND\_VAL, ANL\_DB\_QRY\_ESS, ANL\_MOD\_DAT\_SRC, and ANI\_MOD\_DIM. At the bottom are 'OK' and 'Cancel' buttons.

Change Space dialog box

**Record Name**

Displays the name of the current record definition. This field is display-only.

<b>Platform</b>	Select from the available database platforms to which you can assign the designated space name.
<b>Space Name.DB Name</b> (tablespace name.database name)	Displays the available space name that is assigned to the current record. This field is display-only.
<b>Available Space Name-DB Name</b> (available tablespace name-database name)	Select the space name and database name to assign to the record.
<b>Records Allocated to this Space</b>	Displays all of the records that are already allocated to the space that you selected from the Available Space Name-DB Name drop-down list box. This field is display-only.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," Tablespace Utilities.

## Using Physical Data Storage

You have control over the physical storage of your data. You can view and edit the DDL for creating tables, indexes, and tablespaces in the browser.

Access the DDL Model Defaults page (PeopleTools, Utilities, Administration, DDL Model Defaults.)

DDL Model Defaults

Platform ID:1DB2

Sizing Set:0

Copy...

DDL

Find | View All

First1 of 5Last

Statement Type:Table

\*Model SQL:

CREATE TABLE \*\*OWNER\*\*.[TBNAME] ([TBCOLLIST]) [VOLATILE] IN [DBNAME].[TBSPCNAME];

Parameter Count:1

Parameters

Personalize | Find | View All

First1 of 1Last

DDL ParmDDL Parameter Value

OWNER

DDL Model Defaults page

To view or edit the DDL parameters for creating tables, indexes, and tablespaces:

1. Open the component.
2. Press the Enter key to view platform names.

### 3. Select a platform name.

The DDL Model Defaults page appears.

The Model SQL edit box shows the DDL template for the specified platform and sizing set. The items in square brackets are special parameters that are populated when instances of SQL are generated. For example, the preceding example shows an index model statement for ALLBASE. In this example:

- [TBNAME] is the name of the table.
- [TBCOLLIST] is replaced by the columns that are specified in the index definition.

The model statements also contain parameter names enclosed in pairs of asterisks. A parameter name is replaced by a value when instances of the SQL are generated. The preceding example has one parameter, **\*\*FILESET\*\***. When DDL is generated using this model, the FILESET parameter is replaced by FILE unless it is overridden for the specific record or tablespace. The other text in the model statement is copied to the generated SQL.

### **See Also**

*PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," Using Administration Utilities

## Chapter 8

# Designing Pages for Optimal Performance

This chapter discusses how to improve online performance

---

## Improving Online Performance

To improve the online performance of internet applications, you must optimize the transmissions to the server. Optimizing transmissions improves overall processing speed and reduces a user's data entry time.

Transactions that are designed using deferred mode reduce traffic to the application server but do not operate interactively.

This section provides an overview of processing modes and discusses how to:

- Refresh pages.
- Trigger server processing.
- Use PeopleCode in deferred mode.
- Use page controls in deferred mode.
- Use error and warning messages.

## Understanding Processing Modes

PeopleSoft applications can run in two different processing modes: interactive and deferred.

### ***Interactive Mode***

In interactive mode (formerly called standard), when the user exits a field that has a field-level event (for example, FieldChange, FieldEdit, or RowInit PeopleCode; prompt validation; related display; and so on):

1. A transmission to the application server occurs immediately so that it runs the field-level event.
2. The system refreshes the information.

## Deferred Mode

In deferred mode, the application enables you to temporarily delay many of the conditions that need server processing until you encounter an event or condition that *requires* immediate processing or until you request processing manually. For example, when a user exits a field that has a field-level event (like FieldChange or FieldEdit PeopleCode, prompt validation, related display, and so on), that event is not processed until the next transmission to the application server. When the next transmission to the server occurs, PeopleTools determines which fields have changed since the last transmission. Then, logic on the application server runs the appropriate system edits and PeopleCode events (in field layout order).

You can select deferred mode processing at the field, page, and component levels. For a field in the component to run in deferred mode, you must select deferred mode at each of those levels. For a page to run in deferred mode, you must select deferred mode at the page and component levels. Deferred processing is the default mode at the field, page, and component levels.

## Related Content and Processing Mode

When designing related content, you should carefully review and understand the interaction between the processing mode of any fields, pages, or components to which you associate related content services. This type of careful review will reduce the likelihood that unexpected or contextually irrelevant data appears when the processing mode is incorrect.

You should use deferred processing judiciously when interacting with related content services.

## Refreshing Pages

Click the Refresh button on a page to force processing on the application server to determine which fields have changed since the last transmission and to run processing logic that is associated with those changes.

Users can also refresh by pressing the alt-0 hot key, which keeps the cursor in the same field when the system displays the page. You can refresh at any time during data entry to enable an expert user to:

- Update related display field values for data that is already entered.
- Recalculate totals and balances.
- Provide defaults that are based on data that was entered previously on the page.
- Validate the data that has been entered on the page so far.
- Invoke any hide, unhide, gray, or ungray PeopleCode that is associated with specific fields.

---

**Note.** The Refresh button does not refresh the page from the database. It does cause a server trip so that any deferred PeopleCode changes can be processed. If no deferred changes were made or the deferred changes do not cause any errors or other changes on the page, then clicking Refresh has no effect. The user might think that nothing happened. Fields on derived work records are not updated if the user clicks the Refresh button.

---

## Triggering Server Processing

The following table shows the events that cause the PeopleTools runtime environment to process logic on the application server when operating in deferred or interactive processing mode:

<i><b>Event</b></i>	<i><b>Deferred Processing</b></i>	<i><b>Interactive Processing</b></i>
Entering data in fields with PeopleCode (for example, FieldEdit and FieldChange) associated with them.	NA	X
Entering data in fields that have prompt table edits.	NA	X
Entering data in fields that have related displays.	NA	X
Inserting a row or series of rows in a grid or scroll area.	X	X
Deleting a row from a grid or scroll area.	X	X
Using the grid or scroll area controls to move forward or back.	X	X
Selecting another page tab.	X	X
Selecting another grid tab.	X	X
Expanding or collapsing a collapsible section.	X	X
Clicking any button or icon on the page.	X	X
Clicking any link on the page.	X	X
Clicking the Refresh toolbar button.	X	X
Using the Refresh hot key.	X	X

## Using PeopleCode in Deferred Mode

When designing transactions to run in deferred mode, you:

- Code field edits in FieldEdit PeopleCode; you do not need to move field edits to SaveEdit PeopleCode.

- Code field change logic in FieldChange PeopleCode.
- Set all fields, including those that have FieldEdit or FieldChange PeopleCode, to deferred processing.
- Do not use gray and hide fields on the same page to achieve a form of progressive disclosure on the page.

Specifically, do not include hide or unhide and gray or ungray FieldEdit and FieldChange PeopleCode for definitions on the same page unless the page is triggered by a separate push button. Instead, use either page Activate PeopleCode or FieldChange PeopleCode for definitions that are on another page in the component. The appearance of the page should be static as the user enters values and moves between fields on the page.

You should hide or unhide definitions or set them to display-only in page Activate PeopleCode before the initial appearance of the page based on setup data, configuration options, or personalization settings. Set fields to display-only in PeopleCode by setting the DisplayOnly property in the Field class to *True*.

- Show all fields as a static form and perform cross-validation edits to prevent invalid data combinations from being written to the database.
- For cross-validation edits, always assume that the controlling field (for example, a radio button, check box, drop down list box, and so on) is correct when editing for invalid combinations.

Use SaveEdit to code cross-validation edits. If the transaction requires that the edits be performed before the user can select another folder tab, then code the edits in a function that is called from the page Activate PeopleCode for every other visible page in the component. If the transaction is a sequential process, add the cross-validation edits to the Next and Previous buttons that a user uses to navigate through the process. Code cross-validation edits at the record level if the cross-validation edits apply to all components, and at the component level if they apply to only one component.

### ***Publishing Messages in PeopleCode***

Use the SavePostChange event, either from record-field or component PeopleCode, to publish messages online. PeopleTools has special logic in SavePostChange PeopleCode that defers publishing until just before the commit for the transaction. This PeopleCode minimizes the time that the system locks single-threaded PeopleSoft Integration Broker tables.

### ***See Also***

*PeopleTools 8.52: PeopleCode Developer's Guide*, "PeopleCode and the Component Processor," Deferred Processing Mode

## **Using Page Controls in Deferred Mode**

Consider the following information when you design transaction pages to run in deferred mode.

### ***Edit Tables***

Set fields that have edit table validations to deferred. The user can click the Refresh button for immediate validation of the field; otherwise, the prompt validation is deferred until the next required transmission to the application server.



## ***Prompt Tables***

Use deferred processing for prompt table edits that use %EditTable to set the prompt keys based on the values of other fields on the page. Deferred processing still results in the appropriate behavior being run. For example, if the user clicks the prompt icon, the system:

1. Determines which fields have changed since the previous server transmission.
2. Runs the appropriate PeopleCode events.
3. Applies the values from the page to the buffer.

This action results in setting the correct values before resolving the %EditTable function, provided that any controlling fields come before the prompt table edit based on the page layout order.

4. Runs the prompt table edit lookup logic for the corresponding field.

## ***Drop-Down List Boxes***

Do not use drop-down list boxes when the high-order key values are set on the same page. If you do, it requires the server to transmit the correct drop-down list box values based on the high-order key values. Therefore, set the high-order keys on another page, if possible, or use a prompt lookup instead.

For pages that have an effective-dated scroll area and a drop-down list box based on the PSXLATITEM translate table, the drop-down list box is not refreshed when the effective date is changed on the page. However, the drop-down list box does accurately reflect the new effective date the next time that the page is displayed. Evaluate whether this is a problem for the transaction, based on whether customers typically enter effective-dated XLAT rows for the field and on how often changes to the XLAT values are made. If this is an issue, either replace the drop-down list box with an edit box and a prompt, or use the refresh icon after the effective date field.

Drop-down list boxes are a good design choice because they present users with a small number of selections from which they must select one. They are typically more appropriate than prompts for transactions that are targeted at casual, untrained users. Prompts are more appropriate when a power user is entering the transaction from a paper form or document that contains the necessary codes.

## ***Check Boxes and Radio Buttons***

Remove any FieldChange PeopleCode associated with check boxes in a grid that are mutually exclusive (meaning only one row can be checked). Instead, implement cross-validation edits in SavePreChange PeopleCode. When you design cross-validation edits between a radio button or check box and any associated edit box fields, assume that the controlling field (in this case, the radio button or check box) is correct.

## ***Related Displays***

When you include related displays on your pages, make sure that they are set to deferred mode and that the standard Refresh button is enabled. Then, users can press the Refresh hot key to display related displays when they want to see them. Self-service transactions should not display the code and corresponding related display; they should use descriptive prompts.

## Adding Multiple Rows

This feature enables the user to specify the number of blank rows to be added when clicking the Insert Row button.

The number of blank rows that the user specifies is included on the page when it is refreshed, so application server processing is no longer required to add individual rows.

Do not use this feature for effective-dated grids or scroll areas.

Multiple row inserts might not apply if the entire row is populated using PeopleCode, especially if the data is copied from previous rows. If a multiple row insert does apply, the default value of the `ChangeOnInit` property can be used. (The default value is `True`, which means that any PeopleCode updates done in the `RowInit` or `RowInsert` events set the `IsChanged` and `IsNew` properties to `True`.)

After a transaction is saved, empty rows are discarded before the page is displayed again to the user. An empty row means that the user did not access the data and that PeopleCode or record defaults might have been used to initialize the row for the initial display.

PeopleCode Save processing (that is, `SaveEdit` and `SavePreChange` PeopleCode) runs for all rows that are in the buffer (including the empty ones). Therefore, `SaveEdit` and `SavePreChange` PeopleCode should be coded so that it runs only if the field contains data or if the row properties `IsNew` and `IsChanged` are both true.

An alternative method is adding PeopleCode in the first save program in the component to explicitly delete any row based on the `IsNew` and `IsChanged` properties. If you choose this method, then rows should be deleted from the bottom of the data buffer to the top (last row first).

To implement multiple row insert:

1. Ensure that deferred processing is set.

Open PeopleSoft Application Designer and ensure that deferred mode is set for the component, all pages in the component, and all fields on each page.

2. Access the Page Properties dialog box of the appropriate control.
3. Enable multiple row insert.

For each grid or scroll area where appropriate, select the Allow Multi-row Insert check box under the Use tab in the grid or scroll area property sheet.

4. Add `ChangeOnInit` PeopleCode.

Setting the `ChangeOnInit` property for a rowset to `FALSE` enables PeopleCode to modify data in the row set during `RowInit` and `RowInsert` events without flagging the rows as changed. This functionality ensures that only changes entered by the user cause the affected row to be saved.

---

**Note.** Each rowset that is referenced by a grid or scroll area with multiple row insert enabled should have the `ChangeOnInit` property for the rowset, including child or lower-level rowsets, set to `False`. In addition, you must set this property before any `RowInsert` or `RowInit` PeopleCode for the affected row.

---

## Using Error and Warning Messages

When components, pages, and fields are set to deferred mode, FieldEdit PeopleCode errors and warnings do not appear when a user exits a field; they appear after the next application server transmission. Therefore, users might not receive an error message until they enter all of the data and click the Save button.

For FieldEdit error messages running in deferred mode, the system changes the field to red and positions the cursor at the field in error when it displays the message, enabling the user to associate the error message with a specific field. However, for warning messages, the system does not display fields nor reposition the cursor. Consequently, create warning messages that clearly describe to which fields they apply. For example, if a page contains a "Date out of range" warning, users will be confused if multiple date fields appear on the page.



## Chapter 9

# Using Page Controls

This chapter provides an overview of page controls and discusses how to use:

- Charts.
- Frames.
- Group boxes.
- Horizontal rules.
- Images.
- Static text.
- Check boxes.
- Drop-down list boxes.
- Radio buttons.
- Edit boxes and long edit boxes.
- Pop-up pages.
- Subpages.
- Secondary pages.
- HTML areas.
- Push buttons and links.

---

## Understanding Page Controls

You use three categories of controls for page design and development. You determine which types of controls you want to add to your page definition by considering how you want to organize information on a page, as well as how your users will enter data.

This section discusses:

- Aesthetic controls.
- Data entry controls.
- Function and data processing controls.

## Aesthetic Controls

Use aesthetic controls to help organize, display, or emphasize information on a page. Usually, aesthetic controls are not associated with a particular record field that is defined in a record definition and maintained in a database. With the exception of the image control, these controls never update data in the database. Aesthetic controls include:

<b>Chart</b>	Draws a frame that serves as a placeholder for a chart that you construct using the GetChart field type in PeopleCode. Using the chart control, you draw a box in which the chart appears to the user at runtime. Associate the chart control with a record field in a record definition.
<b>Frame</b>	Draws a display-only box of variable size to visually group a set of controls on a page. You can draw a frame around almost any control. You can isolate controls, such as HTML areas, from other controls on a page and then hide the frame. Frames generate HTML tables for viewing in a browser.
<b>Group Box</b>	Draws a display-only box of variable size with a text label to visually group and identify related controls, such as radio buttons. Like a frame, you can resize a group box to any length or width. However, with a group box you can assign a visible label. Like frames, group boxes generate HTML tables for viewing in a browser.
<b>Horizontal Rule</b>	Draws a horizontal line to separate parts of a page. Use a horizontal rule control as a visual break between controls.
<b>Image</b>	<p>Draws a frame that you can associate with a variable image from a record field in a database. The record field stores a graphic, such as a scanned image of an employee or a picture of an asset, in the format that you defined in the record definition. Because an image can change at runtime and is associated with a record and field definition, it is considered a data entry control and an aesthetic control.</p> <p>The field that you associate with your image can be either a standard Image type field or an ImageReference type field, where images might be associated with it depending on user input. The ImageReference field type is a pointer to an image definition, which enables you to display images dynamically. An example of this is referencing a red, yellow, or green light image definition on a page, depending on the context when the page runs.</p> <p><u>See Chapter 4, "Creating Field Definitions," Specifying Image Field Attributes, page 73.</u></p> <p><u>See Chapter 4, "Creating Field Definitions," Specifying ImageReference Field Attributes, page 73.</u></p>

**Static Image**

Draws a frame that you can associate with an image definition. It can help organize or identify information that appears on a page. A static image is different from an image because it does not change at runtime, and it is not associated with a record field but instead with a predefined image definition.

For example, you might select a static image, such as the yellow Clear button, from the image catalog and place it on top of a push button control so that the user knows to click that button to clear the contents of the page.

See [Chapter 16, "Creating Image Definitions," Image Catalog, page 488.](#)

**Static Text**

Adds a static text item, a display-only alphanumeric field with a maximum length of 60 characters, to describe a page, control, or group of controls.

You might use static text for a static note, an extension of a control label, and occasionally a control label itself. You should avoid using text items because they make it more difficult to translate pages to other languages. Instead, you should reference message catalog text to facilitate translation and other maintenance issues. Other methods of displaying text on a page are display-only edit boxes, field labels, and display-only long edit boxes.

The system disables the static text button when you

## Data Entry Controls

Use data entry controls to offer different ways to enter and maintain information. These types of controls are always associated with a record field that is defined in a record definition and maintained in a database.

---

**Note.** PeopleSoft Application Designer allows you add up to 405 fields on a page. When you reach this limit, typical system behavior includes being unable to select, move, or edit the page fields with an order number exceeding the 405-field limit. While this limit has no effect on the page at runtime, it has the potential to disrupt productivity during design time. If you find a page nearing the 405-field limit, you should use subpages to decrease the number of fields on a single page.

---

See [Chapter 9, "Using Page Controls," Using Subpages, page 236.](#)

**Check Box**

Adds a small square box that operates as a toggle switch—on and off—for data controls that can have one of two values (yes or no, 1 or 0).

During data entry, when the page is saved, the enabled or disabled value (whichever represents the current state of the check box) is written to the database. If values are specified in the page definition, the fields are left blank in the database, but you can still select or deselect the check box on the page. You will not know that the database fields are blank until you run a query or report on the field.

**Drop-Down List Box**

Adds a drop-down list box from which you select a single value from a list of valid values. Use drop-down list boxes to enable data selection from a list of three or more possible choices. In its closed state, the control displays the current value for the control. The user opens the list to change the value.

Drop-down list boxes are a good design choice when presenting users with a small number of selections from which they must select one value.

However, when designing for grids and scroll areas, prompts give you better performance.

**Edit Box**

Adds an edit box, which you use for data entry. You also use edit boxes for displaying fields and translatable text. Use an edit box for text data entry, for example, for a record field that is defined as character, number, signed number, or date.

**Long Edit Box**

Adds a long edit box, which is a variable-length, alphanumeric control used for entering long, textual items, such as comments. The length of the control is determined by its content rather than the physical size of the box on the page.

Use long edit boxes to display long, translatable text. Each long edit box has a built-in scroll bar that enables users to enter and display more data than can be shown at one time on a page. The scroll bar on the long edit box is used only to scroll through the text in the long edit box.

Unlike edit boxes, which are limited to the field size that is defined in your record definition, long edit boxes can contain a relatively unlimited number of characters, up to a theoretical maximum of 64KB. This space, however, is shared by a number of other programming elements, so the true size might be closer to 15 to 25KB, depending on the location of the field in your application and the state of the application.

Typically, you use long edit boxes for comments or descriptions, or you configure the long edit box as a rich text editor.

See [Chapter 11, "Creating Page Definitions," Setting Options Properties, page 356.](#)

**Radio Button**

Adds a small, round button that represents one value for a control with multiple defined values. Add radio buttons in groups. A user can select only one radio button in a group at one time, like station buttons on a radio—hence the name. Use radio buttons to enable selection of one out of two possible choices. If you have three or more choices, use a drop-down list box.

**Subpage**

Adds a predefined, presized group of controls, such as address controls, that are defined in a separate subpage definition. During design time, you add only the subpage control to represent all of the controls in the subpage. You maintain those controls in only one place, the subpage definition. At runtime, you see all of the controls that are defined in the subpage on the page.



## Function and Data Processing Controls

Use function and data processing controls to provide a mechanism for running commands and to maintain levels of information on a page.

<b>HTML Area</b>	<p>Adds an area where you can write your own HTML. With other controls, the PeopleSoft system automatically generates the HTML code and then inserts the HTML code into the dynamically generated code at runtime.</p> <p>See <a href="#">Chapter 9, "Using Page Controls," Using HTML Areas, page 246.</a></p>
<b>Tree</b>	<p>The dynamic tree control is a deprecated feature. Use PeopleCode with the HTML area to create a hierarchical tree.</p>
<b>Push Button/Hyperlink</b>	<p>Adds a push button or link that represents an internal or external link, PeopleCode command, process through PeopleSoft Process Scheduler, prompt action, scroll action, secondary page, or toolbar action. You can specify whether the control appears as a traditional push button or as a link (highlighted, underlined text).</p>
<b>Secondary Page</b>	<p>Adds an invisible control that associates a secondary page with the primary page. You then associate the secondary page with a command push button, link, or a pop-up menu. Secondary pages gather or display supplemental information that is related to the data in a primary page but less frequently referenced or updated. Secondary pages are displayed using the <b>DoModal</b> and <b>DoModalComponent</b> PeopleCode functions.</p> <p>See <a href="#">Chapter 9, "Using Page Controls," Using Secondary Pages, page 239.</a></p> <p>See <i>PeopleTools 8.52: PeopleCode Language Reference</i>, "PeopleCode Built-in Functions," DoModal.</p>
<b>Scroll Area</b>	<p>Provides an easy way for you to group or repeat multiple fields of data in a defined area. Like a grid, users can easily navigate through the rows using links and buttons in a navigation bar, and they can add or delete rows using push buttons. These features are automatically placed in the navigation bar. The navigation bar also provides several other settings, such as a Find feature that enables the user to search all fields and rows for specific data and a View All option so that the user can see all rows of data at once.</p> <p>You can place the fields in the scroll area randomly, one on top of the other, or side by side. Unlike a grid, you are not limited to the type of controls that you can place in a scroll area. You can even place a grid inside a scroll area.</p>
<b>Scroll Bar</b>	<p>Like the scroll area, scroll bars contain push buttons and links for navigation, but not in the form of navigation bars. Developers must manually position all navigation items.</p> <p>Like grids and scroll areas, the actual scroll control that you see in PeopleSoft Application Designer when working with scroll bars in page definitions does not appear at runtime. Instead, the scroll bar control has scroll action buttons to replace the visual rendering of the scroll bar as push buttons and links on the web.</p>

**Grid**

Adds a grid, which looks and behaves like a spreadsheet that is embedded in a page. It has column headings and cells and uses push buttons, links, and tabs to navigate through the data. It is similar to a scroll area on a page. Each row in the grid corresponds to a set of controls in a scroll occurrence. Navigation links and push buttons replace the actual visual scroll bar, and add and delete push buttons enable a user to insert and delete rows. Use instead of a single-level scroll area or scroll bar to manage multi-row sets of data.

---

## Using Charts

Use the Chart Properties dialog box to:

- Associate a chart with a record definition and record field.
- Assign a page field name.
- Set label attributes.
- Set the chart as a page anchor.

On the Label tab, add an information-only name for your chart. This label does not appear at runtime, but it does appear on the control order list on the Order tab of the page definition and any page definition report that you print. Assigning a label is helpful if you have multiple charts on your page.

**See Also**

*PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," GetChart

---

## Using Frames

Use the frame control to visually group a set of controls on a page.

To change frame labels and display options:

1. Double-click a frame to open the Frame Properties dialog box.
2. On the Label tab, enter a brief text description of the function of the controls in the frame.

The default frame label is *Frame*. Any label that you attach to a frame is for information only—it does not appear on your page, but it does appear on the page definition printout and in the control order list. Use labels to differentiate among multiple frames on your page.

For example, if you are enclosing address controls, you might use the label *Address Frame*.

3. Select the style for the frame.

You can control the line color and thickness and the background color of a frame by specifying a style.

See [Chapter 15, "Creating Style Sheet Definitions," page 445](#).

4. Select the Hide Border check box to hide the border of a frame.

Use this option to use the style to shade only the background of the framed area or to apply other styles. If you select Hide Border, it overrides the border options that are specified in a style.

You might also use a frame with a hidden border to facilitate HTML generation. When you insert a frame in a page section, the system generates that section as a table in HTML. This functionality is useful to ensure the correct layout. However, in some browsers, adding more HTML tables can degrade performance.

See [Chapter 8, "Designing Pages for Optimal Performance," Improving Online Performance, page 199](#).

5. Select the Adjust Layout for Hidden Fields check box to enable automatic vertical adjustments to the frame size when hidden fields are present.

If visible fields are not present to the right or left of the hidden field in a frame, the frame collapses to surround the remaining fields.

6. If the field is associated with multicurrency processing, select the Multi-Currency Field option on the Use tab in the Frame Properties dialog box.

This selection causes the field to appear at runtime only if the Multi-Currency option on the PeopleTools Options page is selected.

---

## Using Group Boxes

This section discusses how to:

- Insert and move a group box.
- Set group box properties.

### Inserting and Moving a Group Box

To insert and move a group box:

1. Select Insert, Group Box.
2. Position the mouse where you want the initial location of the group box and click.
3. Move the group box to the appropriate location by clicking and dragging.
4. Adjust the group box size and shape using the handles on the edge of the group box.
5. Remove cursor focus from the group box by clicking anywhere outside of the group box in your page workspace.

## Setting Group Box Properties

Use group boxes to visually associate relevant page field controls and to create a page that is intuitive to the user .

### Setting Group Box Record Properties

Although the group box Record Field properties are optional, associating the group box with a record definition and record field enables you to:

- Distinguish individual group boxes when multiple group boxes appear on the page definition Order tab.
- Use PeopleCode to programmatically control the properties of the group box.

Access the Group Box Properties dialog box. Select the Record tab.

<b>Record Name</b>	Select the record definition to which you would like to associate this group box.
<b>Field Name</b>	Select the field definition to which you would like to associate this group box. Only the fields in the previously selected record definition appear in this list.

### Setting the Group Box Label Properties

Use group box label properties to reflect how the fields that appear inside the group box are related.

However, if for visual reasons the group box surrounds a variety of disparate controls, you might manually enter a label rather than associating the group box with a record definition and field.

Access the Group Box Properties dialog box. Select the Label tab.

1. Select the label Type and select one of these options:

<b>Text</b>	Select to manually enter the group box label. Selecting this option enables the Text field and disables the Label ID field.
<hr/>	
<b>Note.</b> You do not enter the text until the next step.	
<hr/>	
<b>RFT Short</b> (record field text short)	Select to access the Short Name record field label properties of the record field that you associate with this group box. Selecting this option disables the Text field and enables the Label ID field.
<hr/>	
<b>Note.</b> You do not select the label until the next step.	
<hr/>	
<b>RFT Long</b> (record field text long)	Select to access the Long Name record field label properties of the record field that you associate with this group box. Selecting this option disables the Text field and enables the Label ID field.

## 2. Select or enter the specific Label Text.

<b>Text</b>	<p>Enter or view the text that is to appear as the label in the group box header.</p> <p>If you select the RFT Short or RFT Long label type, the system automatically populates this field based on the Label ID that you choose and then disables it.</p> <p>Group box labels are limited to 50 characters, regardless of whether the label is static text or retrieved from the field definition.</p>
<b>Label ID</b>	<p>Select the value that is to appear as the label in the group box header. The values that appear in this list depend on the label type option that you selected in the previous step and the label properties of the field definition. Changing the Label ID value automatically changes the value in the Text field.</p> <p>See <a href="#">Chapter 4, "Creating Field Definitions," Understanding Multiple Labels, page 75.</a></p>
<b>Label Style</b>	<p>Select a value to determine the look and feel of the label.</p> <hr/> <p><b>Note.</b> Not all style classes are appropriate for labels. Changing the default style can produce unexpected results.</p> <hr/> <p>See <a href="#">Chapter 15, "Creating Style Sheet Definitions," Overriding the Default Page Field Styles, page 480.</a></p>

## 3. Select an Embedded Help Pop-up option.

<b>No Popup</b>	<p>Select this option to disable embedded pop-up help.</p> <p>This value is the default.</p>
<b>Message Catalog Popup</b>	<p>Select this option to enable the embedded pop-up help icon on the grid at runtime. Selecting this option enables the Message Set/Number edit boxes.</p>
<b>Message Set/Number</b>	<p>Enter the message set number and message number of the message catalog entry to appear when the user clicks the embedded help icon.</p>
<b>Text</b>	<p>Displays the message that appears when the user clicks the embedded help icon. This field is display-only.</p>

## 4. Select an alignment option for the label.

## 5. Select the Display Options of the label.

### **Body Style**

You can control the color, line thickness, and background of the body section of a group box by specifying a style.

See [Chapter 15, "Creating Style Sheet Definitions," page 445](#).

### **Hide Border**

Select to hide the outside border of the group box. This selection overrides the border options specified in a style.

### **Adjust Layout for Hidden Fields**

Select to have the group box adjust its borders to accommodate any hidden fields.

---

**Note.** For the group box to adjust automatically, the hidden fields inside the group box must be surrounded by open space or have other hidden fields to the right or left.

---

## ***Setting Group Box Use Properties***

Access the Group Box Properties dialog box. Select the Use tab.

### **Multi-Currency Field**

Select to associate this field with multicurrency processing. If the Multi-Currency option on the PeopleTools Options page is selected, the field displays at runtime only.

### **Hide all Fields when Group Box Hidden**

Select to hide all visible and hidden fields when the group box is hidden. Hide a group box by associating it with a record and field and invoking the relevant PeopleCode field class property.

See *PeopleTools 8.52: PeopleCode API Reference*, "Field Class," Visible.

### **Collapsible Data Area**

Select to collapse the group box into a small image that the user must click to expand. Collapsing the group box enables the Default Initial View to Expanded State check box and enables you to select label images for both the expanded and collapsed states, typically small triangles.

Each time that a user opens or closes a collapsible section, a transmission to the application server is required. Therefore, the initial state of whether the collapsible section is open or closed is important. You should carefully evaluate the performance and usability aspects of using collapsible sections versus designing a long page that displays all of the data using deferred processing mode instead.

## ***Example of Collapsible Group Boxes***

In the following example, the first group box, Spain, is expanded while the group boxes for France, Italy, and USA are collapsed:

The screenshot shows a web application interface. At the top, there is a header bar with a dropdown menu showing 'Spain'. Below this is a table with the following columns: 'Effective Date', 'Military Status', and 'Social Security Affiliation Dt'. The table has one row with the value '1' in the 'Effective Date' column. Below the table, there are three links: 'France', 'Italy', and 'USA', each with a small flag icon to its left. The entire section is enclosed in a border, indicating it is a collapsible group box.

Example of collapsible group boxes

## Using Horizontal Rules

This section discusses how to:

- Draw a horizontal rule.
- Set horizontal rule properties.

### Drawing a Horizontal Rule

To draw a horizontal rule:

1. Select Insert, Horizontal Rule.
2. Draw a horizontal line of the appropriate length on the page.

### Setting Horizontal Rule Properties

This section discusses how to apply informational labels to a horizontal rule and configure how the horizontal rule is used on a page.

#### ***Setting Horizontal Rule Label Properties***

To set horizontal rule label properties:

1. Open the Horizontal Rule Properties dialog box.
2. On the Label tab, enter an information-only label to differentiate the line from other frames and lines on your field order list.
3. Select the style for the line.

You can control the color and line thickness of a horizontal rule by specifying a style.

See [Chapter 15, "Creating Style Sheet Definitions," page 445](#).

### ***Setting Horizontal Rule Use Properties***

Access the Horizontal Rule Properties dialog box. Select the Use tab.

**Multi-Currency Field** See [Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348.](#)

**Set to Level 0**

Manipulate the occurs level of your controls on the page while still preserving the physical tab order at runtime. If you adjust the sequence of fields on the Order tab of the page definition, the tab order for the user at runtime is adjusted to that new order.

See [Chapter 11, "Creating Page Definitions," Level-Based Controls, page 314.](#)

Use this setting to specify a scroll level organization that differs from the order of page fields on the Order tab. For example, you might want to create a page that contains multiple collapsing group boxes with each containing level-based controls. By placing a horizontal rule between each of the collapsing sections and selecting Set to Level 0, you return the occurs level to 0 so that you can add additional levels without having to alter the order of controls on the Order tab. In doing so, you can maintain a consistent tab order.

---

## **Using Images**

This section provides an overview of static images and discusses how to:

- Adjust static image size and shape.
- Set static image properties.
- Set image properties.

### **Understanding Static Images**

A static image is different from an image because it:

- Does not change at runtime.
- Is not associated with a record field.

Static images are associated with a predefined image definition. For example, you might select a static image, such as a yellow Clear button, from the image catalog and place it on top of a push button control so that the user knows to click that button to clear the contents of the page.



## Adjusting Static Image Size and Shape

You can adjust the size of a static image by dragging the handles on the edge of the image box , or you can adjust the size in the Static Image Properties dialog box on the Label tab by changing the width and height in pixels.

## Setting Static Image Properties

To associate a static image with an image definition:

1. Open the Static Image Properties dialog box.
2. On the Label tab, select the label type.
3. If you select *Text*, enter a label to identify your static image.

The label does not appear on your page at runtime. Use it to track your images in your scroll order list box and on page definition reports.

4. Select an image ID using one of the following methods:

- Select a predefined image definition from the Image ID drop-down list box.
- Click the Browse button next to Image ID to preview a list of available images.

The Select Image dialog box appears with a list of the image components available. To the right of the list is a preview of the selected image.

5. Click OK to return to the Static Image Properties dialog box.

The image that you selected appears in the image area with its width and height displayed in pixels.

6. Set the format and size attributes for the image.

<b>Scale</b>	Select to scale the image to the size of the frame that you set using the control selection handles or by the width and height that you enter. The scaling occurs when you view the page in a browser.
<b>Size</b>	Select to display the image as is, regardless of the control size on the page.

7. Click OK.

You can now position the static image anywhere on the page.

---

**Note.** You can define your own static images, such as your company logo, by creating a new image definition. After you create the image definition, you can then add your logo to a page using a static image control.

---

See [Chapter 16, "Creating Image Definitions," page 487](#).

## Setting Image Properties

To set image properties:

1. Open the Image Properties dialog box.
2. On the Record tab, select the record and field names with which this image field is associated.
3. Set the image format and size attributes.

<b>Scale</b>	Select to scale the image to the size of the frame that you set using the control selection handles or by the width and height that you enter. The scaling occurs when you view the page in a browser.
<b>Size</b>	Select to display the image as is, regardless of the control size on the page.

4. Specify the alignment of your image.
5. Select a style for the image.

### Setting Image Label Properties

Be sure to assign an informational label to your image. Assigning an informational label to your image is especially important if your application works with screen reader software.

To document an image control with an informational label:

1. Enter a text description in the Alt tag for image region.
2. Select one of the following settings from the Type drop-down list box:

<b><i>None</i></b>	Select if you do not want a label. The Text field becomes unavailable for entry.
<b><i>Text</i></b>	Select to enter a custom label. This value is the default setting.
<b><i>Message Catalog</i></b>	Select to use a preset message from the message catalog. The Message Set and Msg Number (message number) fields become available for entry.
<b><i>RFT Long</i></b> (record field table long) and <b><i>RFT Short</i></b> (record field table short)	Select to use a preset label from the record field set on the Record tab. The Label ID drop-down list box becomes available for entry.

### Setting Image Use Properties

The Use tab of the Image Properties dialog box is the same as for that of an edit box.

See [Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348.](#)

---

## Using Static Text

Access the Static Text Properties dialog box.

### ***Label Text***

#### **Text**

If you select Text, you can use either uppercase or lowercase characters. The text label is limited to 60 characters in length. You can also split your text item into multiple lines. To do this, position the cursor in the Text field where you want the split to occur and click the Insert Line Feed button. A thick vertical bar character appears in the Text field, and, when you close the dialog box, your text item is split into multiple lines.

#### **Message Catalog**

If you select Message Catalog, enter the appropriate message set and number. As with a text label, you are limited to message catalog text of 60 characters.

### ***Style***

You can control the color, font, size, and other characteristics of the text by specifying a style.

See [Chapter 15, "Creating Style Sheet Definitions," page 445](#).

### ***Alignment***

Set the horizontal alignment of your text control, as determined by the left-hand edge of the field. You can align controls to the left-of-center, centered, or to the right-of-center.

### ***Use Tab***

If a text item is associated with multicurrency processing, select the Multi-Currency Field option. This selection causes the text item to appear at runtime only if the Multi-Currency option on the PeopleTools Options page is selected.

---

## Using Check Boxes

When you insert a check box on a page, it appears with a default label, *Dummy Name*, until you specify a record definition name and field name, or a text label.

## Setting Check Box Properties

Define the check box control by assigning the appropriate properties.

### ***Setting Check Box Record Properties***

To link check boxes with associated record definitions:

1. Open the Check Box Properties dialog box.
2. On the Record tab, select the record and field names from the drop-down list boxes.
3. Enter the enabled and disabled values for the check box.

For example, if the enabled value of the check box is *Y* for yes and the check box is selected when the user saves the row, then the *Y* value is written to the database. If the disabled value of the check box is *N* for no and the check box is deselected when the user saves, then the *N* value is written to the database.

The enabled and disabled values that you enter validate against the Translate table. If the system does not find a value, it displays a warning message but allows the value to remain. You might find this functionality useful when you create prototype pages before defining record definitions.

### ***Setting Check Box Label and Use Properties***

After you link the check box to a record definition and field, set the label properties. Otherwise, the default check box label is the field long name. Set how the check box will be used on the Use tab.

#### **See Also**

[Chapter 11, "Creating Page Definitions," Setting Page Field Properties for Controls, page 337](#)

[Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348](#)

---

## **Using Drop-Down List Boxes**

This section discusses how to:

- Set drop-down list box record properties.
- Set drop-down list box label and use properties.

### **Setting Drop-Down List Box Record Properties**

To set drop-down list box record properties:

1. Open the Drop-Down List Box Properties dialog box.
2. On the Record tab, associate the drop-down list box with a record and field.
3. Select a style.

Set the font and color attributes of your drop-down list box data.

See [Chapter 15, "Creating Style Sheet Definitions," page 445](#).

4. Select a displayed text option:
  - *Xlat Short.*
  - *Xlat Long.*
  - *Prompt Table Field.*
5. If you selected *Prompt Table Field*, enter the field name in the Prompt Table Field area.
6. Set the field size for the drop-down list box.

## Setting Drop-Down List Box Label and Use Properties

You can set the label and use properties for a drop-down list box.

### See Also

[Chapter 11, "Creating Page Definitions," Setting Label Properties, page 344](#)

[Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348](#)

---

## Using Radio Buttons

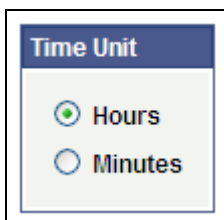
This section provides an overview of radio buttons and discusses how to:

- Set radio button record properties.
- Set radio button label properties.
- Set radio button use properties.

## Understanding Radio Buttons

When you insert a radio button on a page, it appears with the default label, *Dummy Name*, until you specify a record definition name, field, and value for the radio button. Like edit box labels, radio button labels display a dashed box to indicate the translation buffer.

For radio buttons to function as a single group, they must be associated with the same record field and they must be adjacent fields on the Order tab of the page definition. To save time, add all radio buttons in a set, one after the other. After you add the first radio button, the system remembers the record definition name and field name for all subsequent radio buttons; you must enter only the database value for each. Then, place a group box around all of your radio buttons to keep them together and labeled, as in this example:



Example of radio buttons

## Setting Radio Button Record Properties

To link a record with radio buttons:

1. Open the Radio Button Properties dialog box.
2. On the Record tab, select the record and field name.

The system retrieves the default label text if you enter the record or field.

3. Enter the database translate value for this radio button.

In the preceding radio button example, *N* is for *Normal*. If you select the drop-down list box for the value, you see the translate value and the long name that was assigned to the field. When the user selects this radio button, it indicates a *Normal* value.

## Setting Radio Button Label Properties

To set the label of a radio button:

1. Open the Radio Button Properties dialog box.
2. On the Label tab, select the type of label if it is other than the default of *XLAT Long*.
3. Select the radio button label location: to the left or right.

See [Chapter 11, "Creating Page Definitions," Setting Label Properties, page 344.](#)

## Setting Radio Button Use Properties

You might want to change the way that a radio button is used on a page. For example, you might want the selected value of a radio button to control what appears in another control.

See [Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348.](#)

---

## Using Edit Boxes and Long Edit Boxes

When you insert edit boxes and long edit boxes on a page, they appear with the default label, *Dummy Name*, until you specify a record and field for them or until you specify a text type field label. The space between the edit box label and the control is more than required: PeopleSoft Application Designer provides 25 percent extra space for the edit box label for translation purposes.

You control source data, labels, and how the control is used in the respective edit box and long edit box properties dialog boxes in PeopleSoft Application Designer. The settings for these two controls are mostly the same.

---

**Note.** Long edit boxes do not limit the number of characters that a user can enter in the browser, regardless of any limits defined in the underlying record field. As such, characters entered by the user that exceed the field length of the field definition may be truncated. For example, if you enter 012345678901234 into a long edit box associated with a field with a field length of 10, the input would be truncated and saved to the database as 0123456789. The system treats carriage returns (ASCII=13) and line feeds (ASCII=10) the same as any other character.

---

### ***Mouse Over Pop-up Edit Boxes***

You can associate mouse over pop-up pages with edit boxes. The pop-up page appears when you hover your mouse over any edit box that is pop-up enabled. You can identify edit boxes with mouse over pop-up pages associated with them because these edit boxes are both display-only and they appear with a dashed line beneath the control.

You can configure edit boxes to display pop-up pages at runtime by configuring properties on the Use tab at design time.

### ***Display-Only Edit Boxes***

Some edit boxes should be display-only. You might want a page field to govern what appears in another control. In some cases, you might want the page control to be invisible because it is required by PeopleCode but is not accessible by a user.

If you designate the long edit box as display-only, select the Expand Field When Display Only check box to hide the scroll and automatically expand the size of the edit box as necessary to display large amounts of text in the field.

You can configure controls to be display-only at runtime by configuring properties on the Use tab at design time.

### ***Rich Text-Capable Long Edit Boxes***

Some long edit boxes should display the values in rich text format(RTF). RTF is a format type that enables cross-platform document interchange. At runtime, any long edit box that is rich text enabled appears with rows of text formatting toolbars, which enable you to specify formatting information in the text such as font; font size; font characteristics, such as bold and italic; margins; and so on.

You enable long edit boxes to display rich text at runtime by configuring properties on the Options tab at design time.

**See Also**

[Chapter 11, "Creating Page Definitions," Setting Use Properties, page 348](#)

[Chapter 9, "Using Page Controls," Understanding Pop-up Pages, page 226](#)

*PeopleTools 8.52: PeopleCode API Reference*, "Field Class," DisplayOnly

---

## Using Pop-up Pages

This section provides an overview of pop-up pages and discusses how to:

- Define pop-up pages.
- Associate pop-up pages with fields.
- Enable Message Catalog pop-up pages.

## Understanding Pop-up Pages

Pop-up pages are a type of PeopleSoft page that displays information that supplements, expands, or enhances the transaction page data. Pop-up pages are ideal for displaying additional, more detailed information that is pertinent to a subset of users. For example, you can create a Purchase Order Header Details pop-up page that displays customer details such as contact information and payment data. You can then associate the details pop-up page with the Purchase Order Line Items page so that an account representative who is reviewing line items can quickly access customer contact data without having to leave the transaction. Similarly, you can create an Employee Details pop-up page to display an employee's ID, name, department, current job assignment, phone number, and email address. You then associate the Employee Details pop-up page to a specific field on an employee-related transaction page. At runtime, when you move your mouse over this field, the pop-up page appears and displays information that is contextually relevant to the employee data on the transaction page.

Pop-up pages look and behave differently from standard PeopleSoft transaction pages. For example:

- Pop-up pages are not standalone pages; they are linked to, dependent upon, and accessed from their primary transaction pages.
- Pop-up pages do not appear unless you hover your mouse over the page control with which you associated the pop-up page.
- Pop-up pages support only a subset of page controls.
- Pop-up pages cannot contain scroll areas or grids.
- Pop-up page fields are display-only.
- You should place pop-up page PeopleCode either in the page Activate event or on the RowInit event of work records associated with the pop-up page.



- If a value at level 0 on the pop-up page comes from one of the search key fields and if no non-search key value from that record appears at level 0 on the main page, then RowInit PeopleCode in that record does not run.

In addition, if a non-search key value from that record appears on the pop-up page only, then the system does not consider the field to be part of the main page. Because of this, no values for the non-search key fields are retrieved from the database. To populate non-search key fields on the pop-up page with data, you must place a non-search key value from that record on the main page. The field can be hidden on the main page, but it has to exist there.

### ***Designing Context-Sensitive Pop-up Pages***

To create pop-up pages that retrieve data that maintains the key field context of the main page, you must ensure that these conditions are met:

- For each record definition that the pop-up page references, key field values must exist on the main page.
- The names of the main page fields in which these key values exist must be identical to the names of the key fields of the record definitions referenced on the pop-up page.

---

**Note.** While the field names must be identical, they do not have to be key fields in the record definition referenced by the main page.

---

- The fields in which the key values appear must appear in either the same scroll area as the field to which you associate the pop-up page or at a preceding level. The page field to which you link the pop-up page can be one of the fields that contains the key field values; frequently, it is the lowest order key.

Consider this scenario:

If *PRec* is the record that the pop-up page references and the key fields are *PRec.K1* and *PRec.K2*, then, the *K1* and *K2* fields must exist on the main page as *MRec.K1* and *MRec.K2*, where *MRec* is the record that the main page references.

If the field on the main page to which the pop-up page is attached is in *ScrollA*—a level one scroll area—then both *MRec.K1* and *MRec.K2* must either exist in *ScrollA* or at level zero.

---

**Note.** On the main page, it is not necessary that the *K1* and *K2* fields appear in the same record, *MRec* in the previous case. The fields could be *MRec.K1* and *NRec.K2*, where *NRec.K2* is some other field on the main page in the correct context.

---

### ***Context-Sensitive Pop-up Page Example***

In this example; you see a User Details pop-up page. This page is associated with a view, which retrieves information so that the page displays the employee's name, photo, department, phone number, address, and web site URL.



User Details pop-up page

The fields on the pop-up page reference PS\_EMPL\_VW. The key field of PS\_EMPL\_VW is EMPLID, which *requires* that the EMPLID field exist on the main page.

The field to which the pop-up page is linked is at level 0, which *requires* that EMPLID be at level 0 also.

**Note.** EMPLID is a key field on the PS\_EMPL\_VW record that the pop-up page references. While EMPLID also happens to be a key field on the PS\_EMPL\_JOB record that the transaction page references, it is not a requirement.

### Identifying Valid Controls on Pop-up Pages

You may insert these controls only on a pop-up page:

- Frames
- Group boxes
- Horizontal rules
- Static text
- Static images
- Check boxes
- Drop-down list boxes
- Edit boxes

- Images
- Long edit boxes
- Push Button/Hyperlinks
- Radio buttons
- Charts

---

**Note.** Except for Push Button/Hyperlink page controls, these controls are automatically set as display-only at runtime even if you do not set the Display-Only property on the control.

---

### ***Using Push Buttons and Links on Pop-up Pages***

You may use only these destination type values on a pop-up page:

- External link
- Instant messaging action
- Internal link
- PeopleCode command
- Process
- Secondary page

---

**Note.** Do not use other destination types for push button/hyperlink controls on pop-up pages because unexpected results will occur.

---

## **Defining Pop-up Pages**

Before you define your pop-up pages, you should determine which data is appropriate for pop-up information. Typically, this is data that supplements and enhances the data on the primary page or information that you might access from more than one main page.

---

**Note.** When designing pop-up pages, carefully consider how to make them more versatile and reusable.

---

To define pop-up pages:

1. Select File, New and select Page from the New Definition dialog box.
2. Insert the record fields on the pop-up page.
3. Right-click the page and select Page Properties to open the Page Properties dialog box.
4. Select the Use tab.
5. Select *Popup Page* from the drop-down list box in the Page Type group box.

6. Select the page size:

<b>Auto-Size</b>	Select to size the page automatically to fit the fields that are defined on it.
<b>Custom Size</b>	Select to size the pop-up page by dragging the edge or by entering the width and height in pixels.

7. (Optional) Select Adjust Layout for Hidden Fields.

8. Click OK and save your pop-up page.

If you selected Auto-Size, the system sizes the page automatically to fit the fields that are defined on it when you save.

## Associating Pop-up Pages with Fields

To associate pop-up pages with fields:

1. Select File, Open and select the main page.
2. Select Insert, Edit Box or Insert Push Button/Hyperlink..

---

**Note.** You can associate pop-up pages with display-only edit box and push button/hyperlink controls at level 0 or inside of scroll areas only. If you associate pop-up pages with controls that are in grids, you will receive an error.

---

3. Double-click the control.
4. Select the Use tab.
5. Select the Display Only check box.
6. In the Mouse Over Popup group box, select the Page Popup option.
7. Select the name of the pop-up page from the drop-down list box.

---

**Note.** Only pop-up pages appear in the drop-down list.

---

8. Click the OK button.

A dashed line should appear beneath the field to indicate its association with a pop-up page.

9. Save the page.

---

**Note.** If page controls are placed too close together on a page, the dashed line, which indicates an associated mouse over pop-up page, might not appear beneath the control.

---

## Enabling Message Catalog Pop-up Pages

You can enable Message Catalog pop-up pages to embed context sensitive help on any PeopleSoft page. This section discusses how to:

- Enable message catalog pop-up pages.
- Embed context-sensitive PeopleBooks help in pop-up pages.

### ***Enabling Message Catalog Pop-up Pages***

To enable message catalog pop-up pages:

1. Double-click the field that accesses the pop-up page.
2. Select the Use tab.
3. Select the Display-Only check box.

---

**Note.** You must select Display-Only.

---

4. Select the Message Catalog Popup option.
5. Enter the Message Set and Message Number of the message whose Message Text and Explain strings appear in the Message Catalog pop-up page.
6. Click the OK button.

---

**Note.** You can also set the Message Catalog entry for mouse over pop-up pages by using PeopleCode.

---

See *PeopleTools 8.52: PeopleCode API Reference*, "Field Class," MouseOverMsgNum and *PeopleTools 8.52: PeopleCode API Reference*, "Field Class," MouseOverMsgSet.

### ***Embedding Context-Sensitive PeopleBooks Help in Pop-up Pages***

You can add variables to a message to make a Message Catalog pop-up page context-sensitive to the PeopleBooks Online Library.

---

**Note.** You must correctly configure and implement PeopleBooks.

---

See *PeopleBooks and the Online PeopleSoft Library*, Managing the PeopleSoft Online Library and PeopleBooks, Integrating Context Sensitivity.

To add variables to messages to embed context-sensitive help:

1. Access the Message Catalog (PeopleTools, Utilities, Administration, Message Catalog to access the Message Catalog).
2. Select an existing message or create a new message.

3. Enter an explanation and place the variable in the Explanation field.

The table in the next section describes the variables and their usage.

---

**Note.** The text in the Message Text field appears as the pop-up page header. The text in the Explanation field appears in the pop-up page body.

---

4. Enable the pop-up page on the desired edit box control.

See [Chapter 9, "Using Page Controls," Defining Pop-up Pages, page 229.](#)

### ***Embedded Help Variables***

This table describes the variables that you can add to the embedded help content in the Message Catalog. Use these variables to generate links on the embedded help pop-up page that enable the user to access additional online help.

---

**Note.** Your online PeopleBooks library must be installed for any PeopleBooks Online Help link to function properly.

---

<b><i>Variable</i></b>	<b><i>URL</i></b>	<b><i>Label</i></b>
<code>%PBCTXT ( )</code>	This variable uses the current page name as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the current page.	Because no message is referenced, the label of the help link appears as <i>PeopleBooks Help</i> .
<code>%PBCTXT ( sn , mn )</code> The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.	This variable uses the current page name as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the current page.	Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.  If you reference an invalid message, the label appears as <i>PeopleBooks Help</i> .
<code>%PBCTXT ( CTXT_PG_NM )</code> The argument <code>CTXT_PG_NM</code> is a page definition name.	This variable uses the <code>CTXT_PG_NM</code> argument as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the page referenced by the argument.	Because no message is referenced, the label of the help link appears as <i>PeopleBooks Help</i> .
<code>%PBCTXT ( CTXT_PG_NM , sn , mn )</code> The argument <code>CTXT_PG_NM</code> is a page definition name.  The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.	This variable uses the <code>CTXT_PG_NM</code> argument as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the page referenced by the argument.	Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.  If you reference an invalid message, the label appears as <i>PeopleBooks Help</i> .


<b>Variable</b>	<b>URL</b>	<b>Label</b>
<p><code>%PBURLTBL(URL_IDENTIFIER⇒,sn,mn)</code></p> <p>The <code>URL_IDENTIFIER</code> is any <code>URL_ID</code> that is stored in the URL Maintenance component.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Using PeopleTools Utilities," URL Maintenance.</p> <p>The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.</p>	<p>This variable uses the <code>URL_IDENTIFIER</code> argument to build the help URL so that the embedded link takes the user to the site associated with the <code>URL_ID</code>.</p>	<p>Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.</p> <p>If you reference an invalid message, the system uses the <code>URL_IDENTIFIER</code> argument as the label.</p>
<p><code>%PBLNK(http://www.site.com⇒,sn,mn)</code></p> <p>The argument <code>sn</code> is the Set Number. The argument <code>mn</code> is the Message Number.</p>	<p>This variable uses the hard-coded URL so that the embedded link takes the user to the specified site.</p>	<p>Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.</p> <p>If you reference an invalid message, the system uses the URL as the label.</p>

### Example

In this example, the `%PBCTXT` variable uses `QRY_ADMIN` as a `CONTEXT_ID` that references the Query Administration page in PeopleBooks:

## Message Catalog


Message Set Number: 4

Description:  

Short Description:

Messages

Find | View All First 6 of 7 Last

Last Update Timestamp: 07/23/2010 1:45PM  

\*Message Number:

\*Severity:

\*Message Text:  

Description: 

Select this check box to compress queries before sending them to the web server. For more information on this topic, click this link:  
  
%PBCTXT(QRY\_ADMIN,4,7)



Message Catalog page with context-sensitive variable embedded in text

In the Description field, the 4, 7 parameters refer to Message Catalog entry (4,7), which contains the label to use for the generated link.



**Message Catalog**

Message Set Number: 4

Description: Help Processor

Short Description: HPR

Messages		Find   View All	First 7 of 7 Last
Last Update Timestamp: 03/10/2009 9:56AM			
*Message Number:	7		
*Severity:	Message		
*Message Text:	Query Administration		
Explanation:			

Message Catalog page showing Query Administration label in the Message Text field

On the Web Profile Configuration page at runtime, mouse over the selected control to see the generated pop-up page:

**Web Profile Configuration - General**

Profile Name: DEV

Description: Installation Defaults

Authentication Domain:

Help URL: http://peoplebooks.pe

☒ Compress Response

☒ Compress Response

Compress Mime Type

☒ Compress Query ?

Save Confirmation Display Time: 3,000 Milliseconds ?

☒ Enable Processing Message ?

☒ Enable New Window ?

**Compress Queries on the Web Server**

Select this check box to compress queries before sending them to the web server. For more information on this topic, click this link: [Query Administration](#)

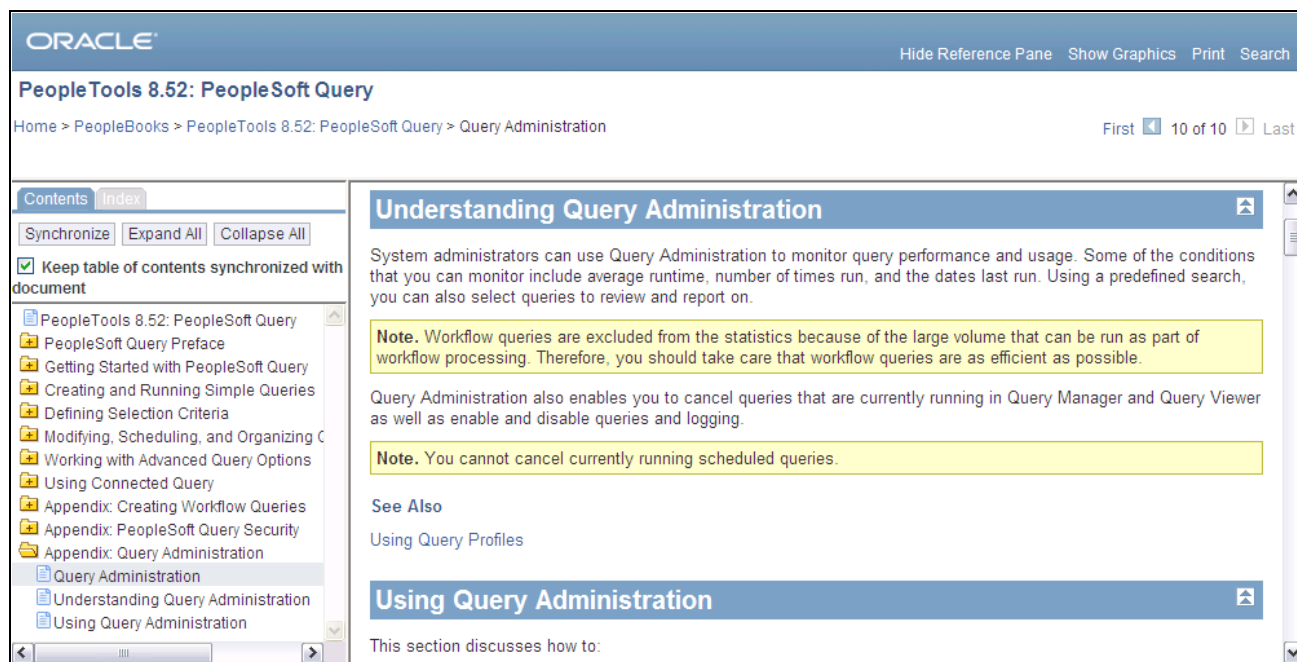
Web Profile - General page showing the message catalog pop-up page

Notice that the pop-up text comes from Message Catalog entry (4,6) and the link label, *Query Administration*, defined by the %PBCTXT(QRY\_ADMIN, 4, 7) function comes from Message Catalog entry 4,7. Click the Query Administration link, and the system accesses the PeopleBooks entry for the *QRY\_ADMIN* CONTEXT\_ID using that Help URL entry to ascertain the PeopleBooks location on the web server.

**Note.** You define the location of your online PeopleBooks by entering a URL in the Help URL field that is located on the Web Profile - General page (PeopleTools, Web Profile, Web Profile Configuration).

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Configuring the Portal Environment," Configuring General Portal Properties.

The result is the PeopleSoft Query PeopleBook topic that you see here:



Query Administration page in the PeopleSoft Query PeopleBook

## See Also

Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Configuring Embedded Help for Scroll Areas and Grids, page 307

## Using Subpages

This section provides an overview of subpages and discusses how to:

- Define subpages.
- Insert subpages into pages.
- Specify informational subpage labels.

## Understanding Subpages

Subpages are a powerful means of factoring out commonly used page functionality in your application. Rather than duplicating the same set of page fields on two or more pages, you can create a single subpage that contains those page fields and then add it to any page. Subpages provide an easy way to maintain the functionality that the page fields represent.

Subpages are even more effective when used with corresponding subrecords. To support the use of subrecords with subpages, the system uses a record name substitution mechanism for subpage-type page fields. Each subpage has a from record that is determined dynamically by the page definition in PeopleSoft Application Designer and by PeopleSoft Pure Internet Architecture. The from record is a record associated with one of the page fields on the subpage. Which page field depends on the page field ordering, the types of page fields on the subpage, and whether the associated record is a subrecord. The subpage itself must be fully expanded to make this determination. (Keep this fact in mind when working with subpages that contain other subpages.)

You specify the to record in the SubPage Record Name Substitution group box in the Insert Subpage dialog box that appears after you insert the subpage into a page. (It also appears on the Subpage tab of the Subpage Properties dialog). At runtime, when a subpage-type page field is expanded, each page field that is associated with the dynamically determined from record is instead associated with the to record that you specify. For example, you can populate a subpage with page fields that are associated with a subrecord so that the subrecord is determined to be the from record; then, when that subpage is used as a subpage-type page field, you can specify the to record as one of the records that contains that subrecord. You are able to associate a single set of page fields (through the subpage) with a variety of different records (each containing the subrecord).

## Defining Subpages

Creating a subpage is like creating a page definition except that you define it as a subpage rather than a standard page in the Page Properties dialog box.

To define a subpage:

1. Select File, New, Page.
2. Insert the record fields on the page.

See [Chapter 11, "Creating Page Definitions," Adding Page Controls, page 329.](#)

3. Select File, Definition Properties to access the Page Properties dialog box.

Use this dialog box to define the type, size, and layout of the page.

4. Select *SubPage* in the Page Type drop-down list box.
5. Select the size of the page.

See [Chapter 11, "Creating Page Definitions," Changing Page Use Information, page 383.](#)

6. Select Adjust Layout for Hidden Fields if necessary.
7. Select Allow Deferred Processing if appropriate.

See [Chapter 8, "Designing Pages for Optimal Performance," Understanding Processing Modes, page 199.](#)

## Inserting Subpages Into Pages

After you create a subpage definition, insert a subpage control on your primary page and associate it with the subpage definition.

To insert a subpage into a page:

1. Open the page into which you want to insert the subpage.
2. Select Insert, SubPage.

The Insert Subpage dialog box appears.

3. Select a subpage definition.

Specify the name of the subpage that you want to insert into the open page definition.

4. Select a record definition in the SubPage Record Name Substitution group box.

The subpage writes its fields to a generic record. Specify the application-specific record name to which you want the information in the subpage written.

See [Chapter 9, "Using Page Controls," Using Subpages, page 236](#).

---

**Note.** All of the fields in a subpage must be associated with fields of a subrecord in the specified record definition.

---

5. Click the OK button.

The subpage appears on the page. Its size reflects the size of the page control group, and it is identified by the subpage definition name. During design time, the fields within the subpage are not visible. At runtime, pages containing subpages are dynamically expanded. Be sure to leave a few pixels of space between the left edge of the main page and the subpage page field to ensure sufficient padding at runtime.

When saving a page with multiple subpages, be aware that you may receive a warning message if duplicate fields are present.

### ***Inserting a Subpage Into a Grid***

You insert a subpage into a grid as you would any other column.

See [Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Inserting and Manipulating Grid Columns, page 283](#).

## Specifying Informational Subpage Labels

Document the purpose of the subpage by changing its informational label. The default subpage label is *Subpage*. Any label that you attach to a subpage is for information only: it does not appear on the page, but it does appear on the page definition printout and in the control order list. Use labels to differentiate among multiple subpages on a page.

To specify informational subpage labels:

1. Open the Subpage Properties dialog box.

Note that the Subpage tab displays the information that you entered on the Insert Subpage dialog box.

2. On the Label tab, enter your informational text label.
3. Enter the page field name in the General tab.
4. (Optional) Select Enable as Page Anchor if you want the subpage to serve as a jump destination on the page.
5. Click OK.

---

## Using Secondary Pages

This section provides an overview of secondary pages and discusses how to:

- Define secondary pages.
- Insert secondary page controls.
- Associate secondary pages with primary pages.


## Understanding Secondary Pages

Secondary pages are pages that you access through another page, usually by clicking a link or push button. Secondary pages are ideal for entering and updating data that is supplemental to the data on the primary page or data that is common to and can be accessed from multiple primary pages. While a secondary page might sometimes appear as just another page to the user at runtime, secondary pages have certain characteristics that are not shared with standard pages. For example:

- Secondary pages are not standalone pages that you access through the menu structure. You can only access a secondary page from its primary page; the primary page and the secondary page are associated by push button or link page controls.
- Secondary pages include OK and Cancel or Return buttons (or equivalent alternatives) so that the user can accept or discard input, dismiss the secondary page, and return to the primary page.
- Secondary pages display no component buttons, such as Save, Return to Search, and so on.

### ***Example: Secondary Page***

In this example, notice the `Change password` link on the General Profile Information page, which is the primary page:

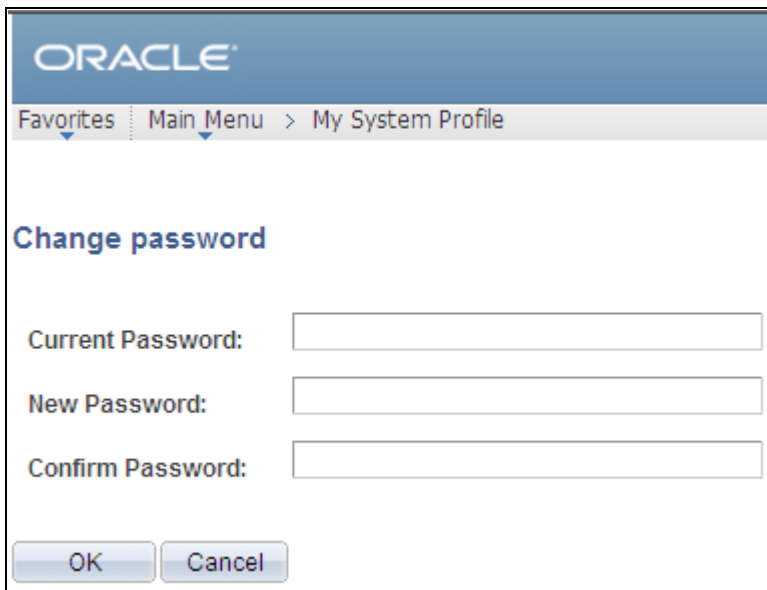


The screenshot shows the Oracle 'My System Profile' page. At the top is the Oracle logo. Below it is a breadcrumb trail: 'Favorites > Main Menu > My System Profile'. The main heading is 'General Profile Information'. Under this heading, it says '[PS] Peoplesoft Superuser'. There is a 'Password' section with a blue bar. Below it are two links: 'Change password' and 'Change or set up forgotten password help'. The next section is 'Personalizations'. It contains several settings: 'My preferred language for PIA web pages is: English', 'My preferred language for reports and email is: English' (with a dropdown arrow), 'Currency Code: USD' (with a search icon), and 'Default Mobile Page:' (with an empty text box and a search icon).

Primary page showing Change password link that opens a secondary page

When the application user clicks the link, it opens the Change password secondary page in the browser.

Secondary pages that open in the browser often are hardly distinguishable from standard pages, as you see in this secondary page example.



The screenshot shows the 'Change password' secondary page. It has the same Oracle logo and breadcrumb trail as the primary page. The heading is 'Change password'. Below it are three text input fields labeled 'Current Password:', 'New Password:', and 'Confirm Password:'. At the bottom are two buttons: 'OK' and 'Cancel'.

Change password secondary page

Click the **Cancel** button to discard any changes that you make on the secondary page. Click the **OK** button to accept any changes that you make on the secondary page. The new information that you enter on the secondary page is not saved when you click the **OK** button; you must save the component.

## Understanding Modal Secondary Pages

Modal secondary pages do not appear in a browser window. Modal secondary pages appear in front of a browser window, capturing focus and disabling the primary page until you dismiss the secondary page.

You can move modal secondary pages by dragging the page title bar. Moving the page is useful when you want to view information on the primary page while working with the secondary page. The appearance of the cursor changes to indicate that you can move the page.

Use these guidelines to determine whether to specify secondary pages as modal. Specify a secondary page as modal when:

- The content of the secondary page is relevant to the context of the primary page.

For example, you need to be able to see both the secondary page and primary page simultaneously.

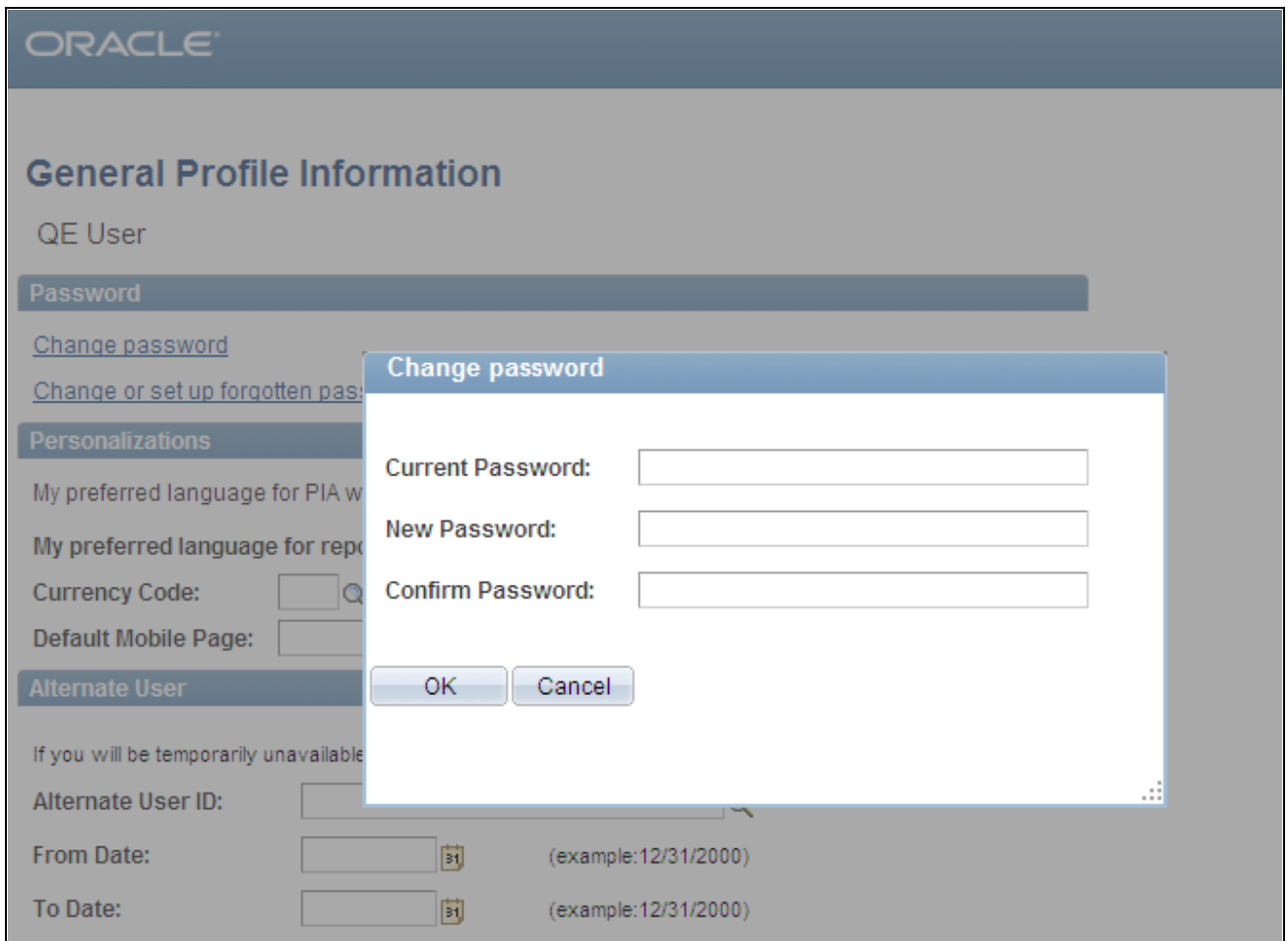
- The content of the secondary page is such that the user can complete the transaction quickly and easily.

By default, secondary pages appear as modal pages. You can change the default behavior at the definition level by deselecting the `Disable Display in Modal Window When Not Launched by DoModal Peoplecode` check box that appears on the `Use` tab on the `Page Properties` dialog box of the secondary page. You also control secondary page modality by using the `DoModal PeopleCode` function.

See [Chapter 9, "Using Page Controls," Defining Secondary Pages, page 242](#).

### ***Example: Modal Secondary Page***

As in the preceding example, the user clicks the `Change password` link to access the `Change password` secondary page. In this example, notice that the `Change password` secondary page is a modal page. It hovers above the `Change Profile Information` page, which is the primary page.

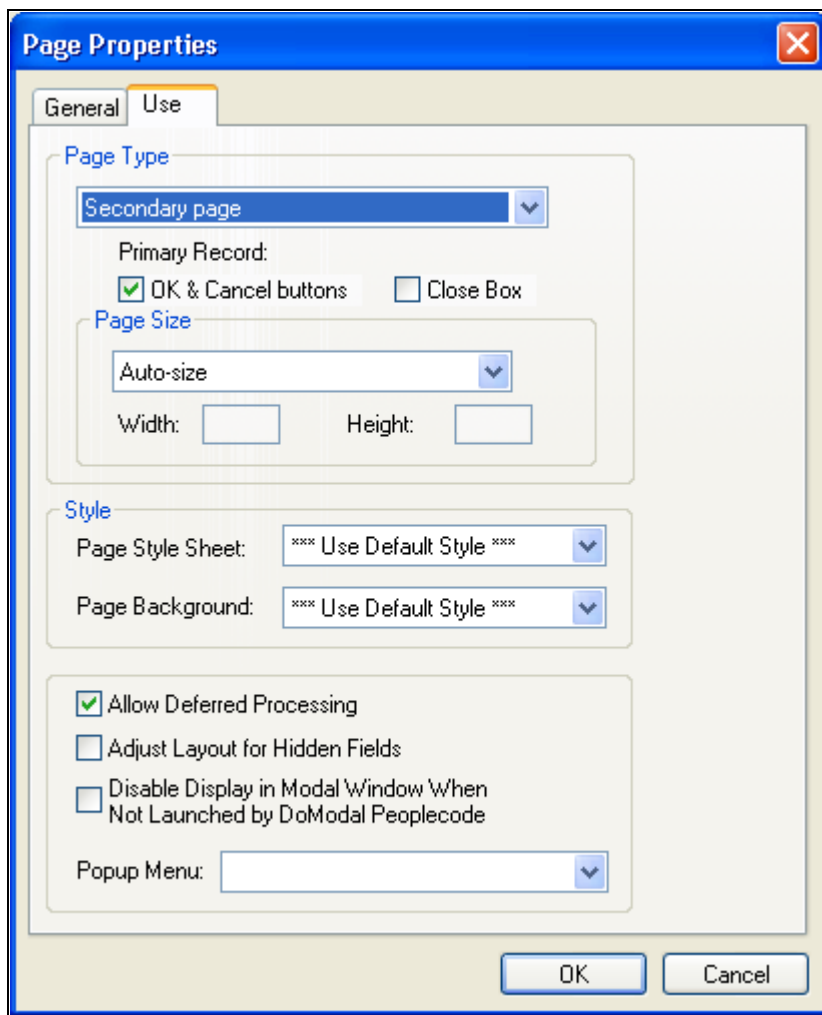


Change password modal secondary page

## Defining Secondary Pages

Access the Page Properties dialog box for the page definition. (With the page open, select File, Definition Properties.)





Page Properties dialog box: Secondary page type

To define secondary pages:

1. Select File, New and select Page from the New Definition dialog box.
2. Right-click the page and select Page Properties to open the Page Properties dialog box.
3. Select the Use tab.
4. Select *Secondary Page* from the Page Type drop-down list box.
5. Deselect the OK & Cancel buttons and the Close Box check boxes if you plan to develop alternative buttons that dismiss the page. To use the standard buttons, accept the default values.

See [Chapter 9, "Using Page Controls," Using Push Buttons and Links, page 250.](#)

6. Select the page size:

**Auto-Size**

Select to have PeopleSoft Application Designer size the page to accommodate the page controls that are defined therein. The right and bottom edges of the page move left and up, respectively, to eliminate any vacant space at the right side and bottom of the page. The top and left edges of the page are stationary, leaving any vacant space at the right side and bottom of the page.

**Custom Size**

Select to manually size the secondary page by dragging the edges or by entering the width and height in pixels.

7. Select the page style sheet to associate with the secondary page.

To override the application default style sheet, select a different style sheet from the drop-down list box. The style sheet that you select is available only for the controls on the secondary page.

8. Select the page background.

You can control the background of the page and any controls that do not have a style sheet associated with them by specifying a style in the page background.

9. (Optional) Deselect Allow Deferred Processing if you want processing to occur each time the user presses the Tab key to advance through the page controls.
10. (Optional) Select Adjust Layout for Hidden Fields if you want the page borders to adjust automatically to accommodate fields that are hidden.

---

**Note.** For the page to adjust automatically, the hidden fields must be surrounded by open space or have other hidden fields to the right or left.

---

11. (Optional) Select Disable Display in Modal Window When Not Launched by DoModal Peoplecode if you want the page to appear within the browser frame comparable to the standard application pages. If you select this option, you must use DoModal PeopleCode if you want the secondary page to appear as a modal page. Leave this check box deselected if you want the secondary page definition itself (not PeopleCode) to control the appearance.

---

**Note.** This option is disabled when the page type is set to *Popup page*.

---

12. Click the OK button.

The page definition size automatically adjusts if you set the page size to *Auto-Size*.

13. Save your secondary page.

## Inserting Secondary Page Controls

After you define a secondary page, you can place a secondary page control on your primary page and associate it with the secondary page you just created.

To insert a secondary page control, access the primary page:

1. Select Insert, Secondary Page.

2. Click the primary page where you want to display the secondary page control.

The secondary page control looks like a small push button with the secondary page icon on it. This control is invisible at runtime.

3. Double-click the control to access the Secondary Page Properties dialog box: Secondary Page tab.
4. Select the name of the secondary page to associate with the secondary page control on the active page.

Only secondary pages can be associated with secondary page controls. You cannot associate a standard page or subpage with the secondary page control.

5. Select the Label tab.

This label is for informational purposes only. It appears in the Order tab of the page definition and in the page description printout.

6. Position the secondary page control at the proper scroll level on the Order tab.

See [Chapter 9, "Using Page Controls," Specifying Type Properties for Push Buttons or Links, page 252.](#)

7. Save the page definition.

---

**Note.** When you use secondary page controls on primary pages, you must also insert a push button control and write PeopleCode that invokes the secondary page.

---

### **See Also**

[Chapter 9, "Using Page Controls," Associating Secondary Pages With Primary Pages, page 245](#)

## **Associating Secondary Pages With Primary Pages**

You can use two page controls to associate a secondary page with a primary page:

- Insert a push button or link page control on the primary page.

When you use this control type, you must set the Type properties to invoke the secondary page. At runtime, when the user clicks the push button or link, the secondary page appears. Use this control when:

- You need no procedural PeopleCode logic before the secondary page appears (PeopleCode can be used on the secondary page just like any other page).
- You want to control the formatting of the information (therefore, you want to use a page and not use the Prompt function).
- You use the secondary page multiple times.

See [Chapter 9, "Using Page Controls," Using Push Buttons and Links, page 250.](#)

- Insert a secondary page control on the primary page.

This control looks like a command push button, but it is invisible at runtime. When you use a secondary page control, you must also insert and configure a push button on the primary page. After you set the push button Type properties to invoke Command PeopleCode, you then call the **DoModal** PeopleCode function from the FieldChange event for the push button to display the secondary page.

See [Chapter 9, "Using Page Controls," Inserting Secondary Page Controls, page 244.](#)

See [Chapter 9, "Using Page Controls," Using Push Buttons and Links, page 250.](#)

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions."

---

## Using HTML Areas

This section provides an overview of HTML area controls and discusses how to:

- Insert HTML areas.
- Populate HTML areas.
- Change HTML area labels.

## Understanding HTML Area Controls

You can insert an HTML area control on any PeopleSoft page. You can insert it at any level on a page, and you can place it in a grid control. This control is rectangular and is easy to resize.

Populate the HTML area control in one of these ways:

- Statically, in the page field property sheet.
- Dynamically, by associating the control with a record field or HTML definition.

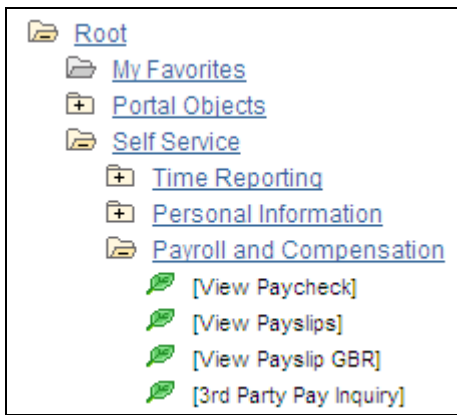
If the control is linked to a record field, the value of the record field appears in the HTML area. Use PeopleCode to associate the HTML area control with a predefined HTML definition.

When using HTML areas, consider:

- HTML areas are downloaded to Microsoft Excel as a nonbreaking space (&nbsp;).
- HTML areas are not searched with the Find feature.

### ***Generating Trees in HTML Areas***

You can use the GenerateTree PeopleCode function with HTML areas. This example shows the tree that is created by the GenerateTree PeopleCode function on the Select New Parent Folder page:



Example of a tree in an HTML area

### Populating HTML Areas

You can populate an HTML area either statically, using the HTML Area Properties dialog box, or dynamically, by associating the control with a record field. Because the HTML that you write is included in the HTML that is dynamically generated by the system at runtime, consider the following:

- The HTML that you include can affect the page layout.

Complying with the design-time sizing of the HTML area is the best way to ensure that you do not affect the layout of the other page field controls. Adding an invisible frame around the HTML area control can help ensure that you do not affect other page fields.

- You can use only certain types of HTML tags. These tags are *not* supported by the HTML area control:
  - <body>
  - <frame>
  - <frameset>
  - <form>
  - <head>
  - <html>
  - <meta>
  - <title>

### Using JavaScript in HTML Areas

If an HTML area contains a reference to third-party JavaScript, you need to include the `absolutized` attribute in the script tag and set it to `false`. For example:

```
<script type="text/javascript" language="javascript" absolutized="false"
  src="../../../ps/spellcheck/lf/spch.js">
</script>
```

### Using Rich Text Editor-Generated HTML in an HTML Area

When you use the rich text editor (RTE) to generate HTML that appears in an HTML area page control, you must call the `ProcessRTEHTML` function to process the RTE-generated HTML before you assign it to the HTML area. The `ProcessRTEHTML` function wraps the HTML inside a `<div>` element, sets the style class to `PT_RTE_DISPLAYONLY`, and attaches the image processing JavaScript. Use the following code as an example:

```
Declare Function ProcessRTEHTML PeopleCode WEBLIB_PTRTE.ISCRIPT1 FieldFormula;

&HTMLAAREA.value = ProcessRTEHTML ("URL ID of the image target", "HTML Data To be⇒
  Processed");
```

Your code might look like this:

```
&HTMLAAREA.value = ProcessRTEHTML (URL.PT_RTE_IMG_DB_LOC, "<p>example</p>");
```

The `&HTMLAAREA.value` should be similar to this:

```
<div id="RTEDiv188" class="PT_RTE_DISPLAYONLY">
  <p>example</p>
  <script type="text/javascript" language="JavaScript">
    PTRTE_CheckImages(" ", "PT_RTE_IMG_DB_LOC", "RTEDiv188");
  </script>
</div>
```

### Using HTML Areas in AJAX Mode

Note the following code restrictions in HTML areas when running the application in AJAX mode:

- Do not include XML tags, such as `CDATA`.
- Do not include empty or commented out `<script>` elements.
- Do not include `document.write` as part of the HTML in an HTML area.

#### See Also

*PeopleTools 8.52: PeopleCode Developer's Guide*, "Using HTML Trees and the `GenerateTree` Function," Using the `GenerateTree` Function

## Inserting HTML Areas

To insert an HTML area on a page:

1. Select Insert, HTML Area.
2. Draw the HTML area on your page.
3. Move the HTML area control by dragging it with the mouse or by pressing the arrow keys.

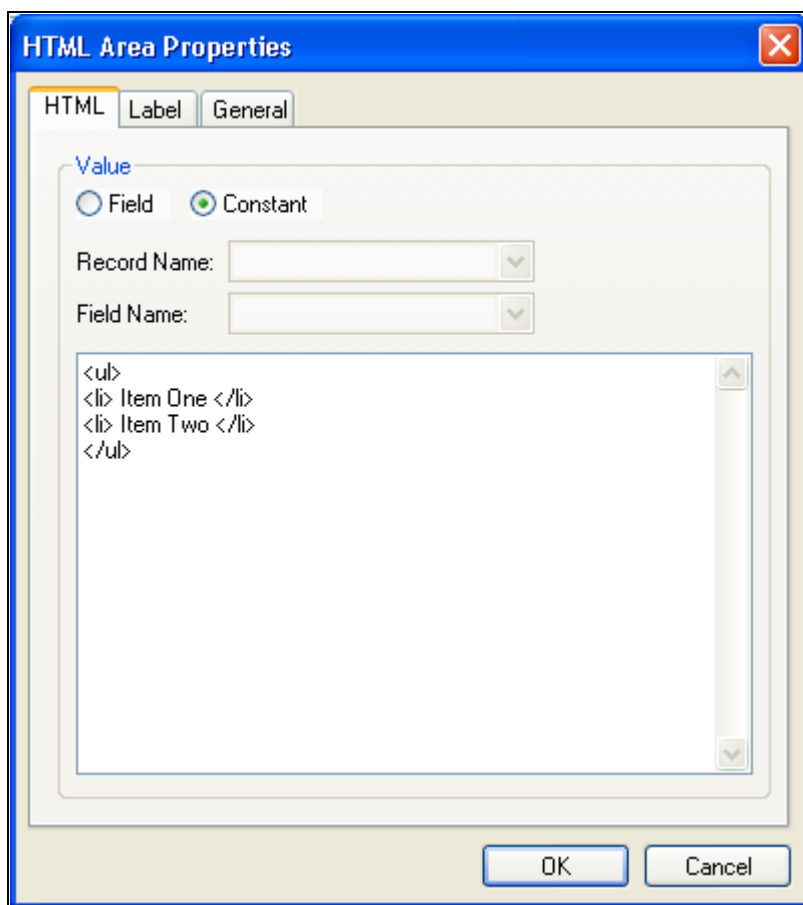
## Populating HTML Areas

This section discusses how to:

- Populate an HTML area statically.
- Populate an HTML area dynamically.

### *Populating an HTML Area Statically*

Use the HTML Area Properties dialog box to populate an HTML area:



HTML Area Properties dialog box with constant text

To populate an HTML area statically:

1. Access the HTML Area Properties dialog box.
2. On the HTML tab, select Constant as the value type.
3. In the long edit box, enter the HTML code that you want to appear in the HTML area.

### ***Populating an HTML Area Dynamically***

To populate an HTML area dynamically:

1. Access the HTML Area Properties dialog box.
2. On the HTML tab, select Field as the value type.
3. Specify the record and field to which you want to associate the HTML area control.

The value of the record field generates the HTML code that is included at runtime in the HTML area.

---

**Note.** When you associate an HTML area control with a field, make sure that the field is long enough to contain the data that you want to pass to it. For example, if you associate an HTML area control with a field that is only 10 characters long, only the first 10 characters of your text will appear. You should use long character fields for record fields that are associated with an HTML area control.

---

## **Changing HTML Area Labels**

Assign a distinct label to your HTML area to distinguish it from other HTML area controls on the Order tab of the page.

To change an HTML area label:

1. Access the HTML Area Properties dialog box.
2. Select the Label tab.
3. Enter a brief text description of the HTML area.

This label does not display at runtime; however, it appears on the Order tab of the page definition.

---

## **Using Push Buttons and Links**

This section provides an overview of push buttons and links and discusses how to:

- Insert push buttons or links.
- Specify type properties for push buttons or links.
- Specify label properties for push buttons or links.
- Specify use properties for push button or link.
- Programmatically change hover text for push buttons or links.

## **Understanding Push Buttons and Links**

PeopleSoft Pure Internet Architecture supports these features for push button and link controls:



- Selecting link style.
- Selecting these types of push buttons and links:
  - External link.
  - Instant messaging action.
  - Internal link.
  - Page anchor.
  - PeopleCode command.
  - Process.
  - Prompt action.
  - Scroll action.
  - Secondary page.
  - Toolbar action.
- Selecting these image options:
  - On mouse over.
  - When disabled.

## Inserting Push Buttons or Links

After you add a push button or link control to your page, you can define the attributes by opening the Push Button/Hyperlink Properties dialog box.

To insert a push button or link:

1. Select Insert, Push Button/Hyperlink.
2. Click where you want the upper, left-hand corner of the control to begin.

A push button of *small image* size appears.

3. Double-click the push button to access the Push Button/Hyperlink Properties dialog box.
4. On the Type tab, select whether the control appears as a push button or link.
5. Select a destination type from the Destination drop-down list box.

See [Chapter 9, "Using Page Controls," Specifying Type Properties for Push Buttons or Links, page 252.](#)

6. Select the record and field names with which you want to associate the push button or link.
7. Select any of the remaining Type options based on your destination type.

8. On the Use tab, select the purpose of the control.

See [Chapter 9, "Using Page Controls," Specifying Use Properties for Push Buttons or Links, page 260.](#)

9. Save the page.

## Specifying Type Properties for Push Buttons or Links

Access the Push Button/Hyperlink Properties dialog box.

The screenshot shows the 'Push Button/Hyperlink Properties' dialog box with the 'Type' tab selected. The dialog has four tabs: 'Type', 'Label', 'Use', and 'General'. The 'Type' tab is active and contains several sections:

- Type:** Two radio buttons, 'Push Button' and 'Hyperlink'. 'Hyperlink' is selected.
- Destination:** A dropdown menu showing 'Internal Link'.
- Record Name:** A dropdown menu.
- Field Name:** A dropdown menu.
- Style:** A dropdown menu showing '\*\*\* Use Default Style \*\*\*'.
- External Link:** Two radio buttons, 'Static' and 'Dynamic'. 'Static' is selected. Below it is a 'URL ID' dropdown menu and a checkbox for 'URL Encoded by Application'.
- Internal Link:** A section with several dropdown menus: 'Portal' (showing 'Use Current'), 'Node', 'Menu', 'Component', 'Market', 'Page', and 'Action'. At the bottom of this section is a checkbox for 'Use data from current page in search'.
- Actions:** Two dropdown menus, 'Action Type' and 'Related Control'.
- Process:** Two dropdown menus, 'Type' and 'Name'.
- Secondary Page:** A dropdown menu for 'Page'.

At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

Push Button/Hyperlink Properties dialog box: Type tab

### Type Tab

Each destination type that you select enables a different region on the Type tab, enabling you to further define your push button or link. Select one of these values from the Destination drop-down list box:

***External Link***

Accesses a uniform resource locator (URL). You can select a value from the URL table or use the value of a record field to define the destination.

Selecting this option enables the External Link group box. You can choose to have the URL encoded or not by selecting the Encoded by Application check box.

Selecting this option applies an external link to your page in the form of a push button or link that accesses a new page, taking the user to the external web site that you designate. To return to the primary page, the user closes the new page. When setting the record and field name, consider these points:

- If you specify a dynamic external link, you must enter a record name and field name.

The system uses the value of the record and field specified as the value of the link.

- Whether the external link is static or dynamic, you can use the specified record and field names to control the page field in PeopleCode when you want, for example, to hide the link dynamically with PeopleCode.
- The tool tip, or mouse over text, that appears with the link at runtime comes from the RFT long label assigned to the record field.

However, the RFT long label appears only if it is different from the assigned display value of the link and it is not null. If the link appears as an image button, the tool tip is derived from the label text if it exists. Otherwise, the system uses the RFT long label.

***Internal Link***

Accesses a PeopleSoft page. Parameters include the menu path to the page and a check box to specify whether data from the current page should be used in the search dialog box.

Selecting this option enables the Internal Link group box.

Selecting this option adds an internal link in the form of a push button or link that you can use to access another component or page from within the system or another PeopleSoft database or node. You can even specify the type of action mode to open in for that page. To control the push button or link in PeopleCode, enter the record and names.

Like the external link, if you associate a record and field name with this link, the tool tip that appears at runtime draws from the RFT long label for the record field. However, the RFT long label only appears if it is different from the assigned display value of the link.

***Instant Messaging Action***

Select to add an instant messaging button to the page. Associate the button with a record field that contains entries in the form of <user>@network (such as John@Yahoo). For this purpose, we provide the MCFIMNETWORKS table. The MCFIMDOMAIN field is the source for the instant messaging network providers. This table contains only three network providers: AOL, Yahoo, and SameTime Connect. You should never enter additional networks not already in this table.

See *PeopleTools 8.52 : MultiChannel Framework*, "Configuring Instant Messaging in PeopleSoft MultiChannel Framework."

***Page Anchor***

Enables the user to jump to the anchor tag that you designate for a page field on the General tab of the properties dialog box for that field.

Selecting this option enables the user to jump from one destination (the link that you assign as the page anchor) to another (the field that you enable as the page anchor) in your page. When you select this option, the page fields that you previously set as page anchors on the General tab of the properties dialog box appear in the Related Control drop-down list box in the Actions group box. You can place as many page anchor buttons or links on the page as necessary.

***PeopleCode Command***

Associates a field in a record, so when the user clicks the push button or link, the system runs any FieldChange PeopleCode that is associated with that field. Enter the record and field names. The field should have PeopleCode in the FieldChange event.

***Process***

Runs processes that you set up previously with PeopleSoft Process Scheduler. Associate process push buttons with a specific process definition; the process is run each time that a user clicks the button.

Selecting this value enables the Process group box, where you can specify the type and name of the process that you want to run.

See *PeopleTools 8.52: PeopleSoft Process Scheduler*, "Submitting and Scheduling Process Requests."

***Prompt Action***

Displays a prompt dialog box for a specific control field. Adds a custom prompt button or link next to a field on the page in place of the standard prompt that is associated with that field. For example, you might want the prompt to read *Find an Airport Code*.

Selecting this value enables the Actions group box. In the Related Control drop-down list box, select the field that you want to associate with the prompt. The system sets the action type, *Prompt*, automatically. Set the text label properties for the link on the Label tab.

***Scroll Action***

Represents certain actions that a user can perform on a grid, scroll area, and scroll bar, such as bottom, top, insert row, and so on.

Use to provide action buttons for the grid, scroll area, or scroll bar in areas outside of that control, such as at the bottom of the page or outside the borders of the control.

Selecting this value enables the Actions group box, where you can specify the action type and related control. The Action Type field enables you to select the action to be performed in the specific level. The level is specified in the Related Control field. Specify one of the following scroll actions: *Bottom, Next, Previous, Row Delete, Row Insert, and Top.*

For example, if you want a user to be able to move through a page at runtime, set the action type to *Next* or *Previous* and specify which scroll area you want those actions to control.

***Secondary Page***

Access an existing secondary page. After you design a secondary page, you must associate it with a control on the primary page. The secondary page push button or link is the starting point for the user of that secondary page. Selecting this option enables the Secondary Page group box, where you can select the name of the secondary page that you want to associate with the push button or link.

***Toolbar Action***

Represents the various toolbar actions available to the user. A user can perform several actions on a page, such as save, display the next page in a group, correction mode, and so on. These normally reside on the toolbar at the bottom of each PeopleSoft Pure Internet Architecture page. Use this type of push button or link for placing save, next in list, or other toolbar functions on the page.

Selecting this value enables the Actions group box, where you can specify the action type: *Add, Correction, Next in List, Next in Worklist, Next Page, Notify, Previous in List, Previous in Worklist, Previous Page, Refresh, Return to List, Save, Update/Display, Update/Display All, or View Worklist.*

***External Link Group Box***

In this group box, select Dynamic or Static and enter the URL ID, if necessary. Select a value from the URL Maintenance table of a record field to define the destination. For example, a derived record field might be used to set the destination value dynamically at runtime. The value of the derived record field is placed in the <HREF> tag of the HTML.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," URL Maintenance.

***Internal Link******Portal***

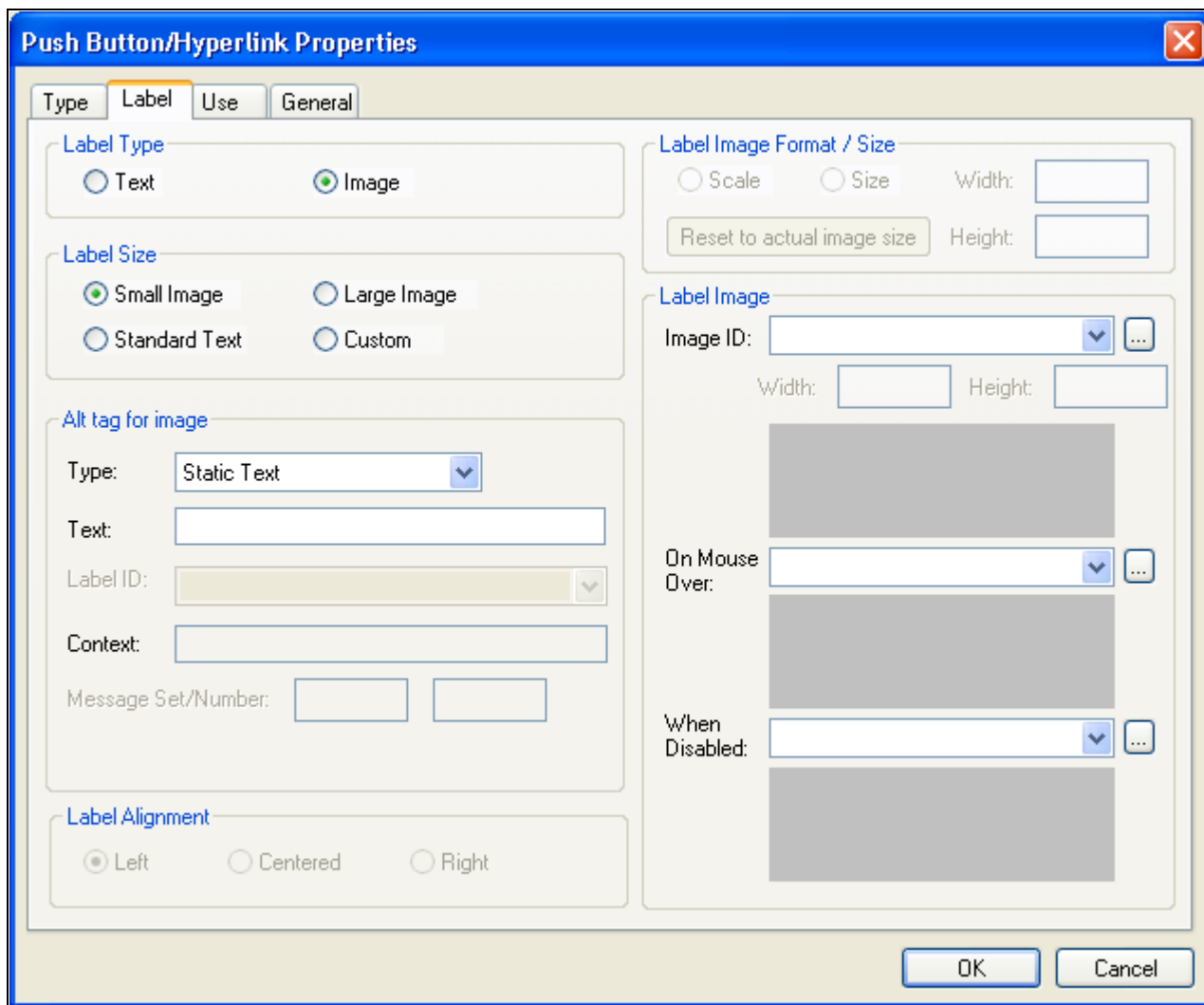
Select the portal to which you want the internal link to point. *Use Current* is the default. Valid values include all of the values in the PORTAL\_NAME column of the PSPRDMDEFN table.

<b>Node</b>	<p>Select the PeopleSoft database to which you want the internal link to point. <i>Local</i> stands for the node that is marked as the default only. If the content reference you need is registered with a non-default local node, make sure you choose the exact same value from this drop-down list box.</p> <p>Valid values include all of the values in the MSGNODENAME column of the PSMSGNODEDEFN table. If you select a node other than <i>Local</i>, the Use data from current page in search check box is deselected.</p> <p>The node that you select should match the node for the destination component in the Registration Wizard or on the Content Ref Administration page. Select PeopleTools, Portal, Structure and Content. Click the Edit link to access the Content Ref Administration page.</p>
<b>Menu</b>	Select the menu name in which the destination page is contained.
<b>Component</b>	Select the component in which the destination page is contained. These values depend on the menu that you select.
<b>Market</b>	<p>Select the market of the destination component.</p> <p>See <a href="#">Chapter 12, "Creating Component Definitions," Specifying Market-Specific Components, page 402.</a></p>
<b>Page</b>	Select the page that should be active. These values depend on the component that you select.
<b>Action</b>	Select the action mode in which you want the page to begin.
<b>Use data from current page in search</b>	<p>Select if you want data from the current page to be used in the search dialog box. This option ensures that the new page inherits the proper keys from the context of the current page. The process is almost identical to the <b>Transfer( )</b> PeopleCode function, and it performs a similar function.</p> <p>If this check box is selected, the system discards the existing keylist; builds a new keylist from the current component buffer using the context of the field (when a choice exists between rows in a scroll area); and uses this keylist to start the new component.</p> <p>If this option is deselected, the system uses the existing keylist in the normal manner.</p>

## Specifying Label Properties for Push Buttons or Links

This section discusses how to:

- Specify a text label.
- Specify an image label.



Push Button/Hyperlink Properties dialog box: Label tab

### ***Specifying a Text Label***

To specify a text label for a push button or link:

1. Access the Push Button/Hyperlink Properties dialog box.
2. Select the Label tab.
3. In the Type group box, select Text.

## 4. Select the size.

Standard push buttons can be one of three sizes, depending on what kind of text or image you want to display on the button face.

<b>Small Image</b>	Creates toolbar-sized buttons that handles small 16 × 16 pixel images.
<b>Large Image</b>	Handles 32 × 32 pixel images.
<b>Standard Text</b>	Matches the Microsoft standard dimensions for text push buttons.
<b>Custom</b>	Select if none of these sizes meets your needs. When you click OK, the push button displays selection handles that you can use to resize it.

## 5. Specify the type of label text.

Depending on the destination type that you selected, the values in the Type drop-down list box differ.

<b>Message Catalog</b>	Select to reference a message that exists in the message catalog. Enter the message set and number. Message catalog items are all translatable.
<b>RFT Short</b> (record field table short)	Select to label the push button with the short name for the field from a record definition. The RFT stores attributes for fields in a record definition. Use this option only if you specified a record definition and field name on the Type tab in the Push Button/Hyperlink Properties dialog box. The RFT short name appears on the push button when you click OK and return to the page. Select the label ID if you do not want to use the default label if more than one label has been set for the record field.
<b>RFT Long</b> (record field table long)	<p>Select to label the push button with the long name for the field. Use this option only if you specified a record definition and field name on the Type tab in the Push Button/Hyperlink Properties dialog box. The RFT long name appears on the push button when you click OK and return to the page. Select a different label ID if you do not want the default label if more than one label has been set for the record field.</p> <p>The system uses the RFT long label as the tool tip at runtime for links if it is different from the assigned display value of the link and it is not null. If more than one label is stored for the record field, you can select an alternate tool tip from the Label ID drop-down list box.</p>
<b>Static Text</b>	Select to enter the text to appear on the label.
<b>URL Description</b>	<p>Select if you selected <i>External Link</i> as the destination on the Type tab. Select the style from the drop-down list box.</p> <p>See <a href="#">Chapter 15, "Creating Style Sheet Definitions," page 445.</a></p>



6. Select the alignment.

This option is only available if you selected Hyperlink for the type on the Type tab. Push button text is automatically centered for controls that appear as push buttons. Push buttons with text and images are automatically center-aligned when used in grids.

---

**Note.** Some browsers do not support customized push buttons.

---

See *Troubleshooting Browser Limitations* available on the My Oracle Support website. [support.oracle.com](http://support.oracle.com).

### ***Specifying an Image Label***

To specify an image label for your push button or link:

1. Access the Push Button/Hyperlink Properties dialog box.
2. Select the Label tab.
3. Select Image for the type.
4. Specify the size.

If you select Custom, the Image/Format Size group box becomes available.

<b>Scale</b>	Select to adjust the width and height as percentages.
<b>Size</b>	Select to set the image size by altering the width and height in pixels.
<b>Reset to actual image size</b>	Resizes the image according to its natural size. That is, the image size is not manipulated by custom width or height values.

5. Specify the image ID.

Select one of the predefined image definitions or click the Browse button to view push button images.

See [Chapter 16, "Creating Image Definitions," Understanding Image Definitions, page 487](#).

6. In the Alt tag for image group box, specify the hover text type.

<b>Message Catalog</b>	Select to reference a message that exists in the message catalog. Enter the message set and number. Message catalog items are all translatable.
<b>RFT Long</b> (record field table long)	<p>Select to use the long name of the record field as the hover text. Use this option only if you specified a record definition and field name on the Type tab in the Push Button/Hyperlink Properties dialog box.</p> <p>The system uses the RFT long label as the tool tip at runtime for links if it is different from the assigned display value of the link and it is not null. If more than one label is stored for the record field, you can select alternate hover text from the Label ID drop-down list box.</p>
<b>RFT Short</b> (record field table short)	Select to use the short name of the field as the hover text. Use this option only if you specified a record definition and field name on the Type tab in the Push Button/Hyperlink Properties dialog box. If more than one label is stored for the record field, you can select alternate hover text from the Label ID drop-down list box.
<b>Static Text</b>	Select to enter the text that will appear on the label.

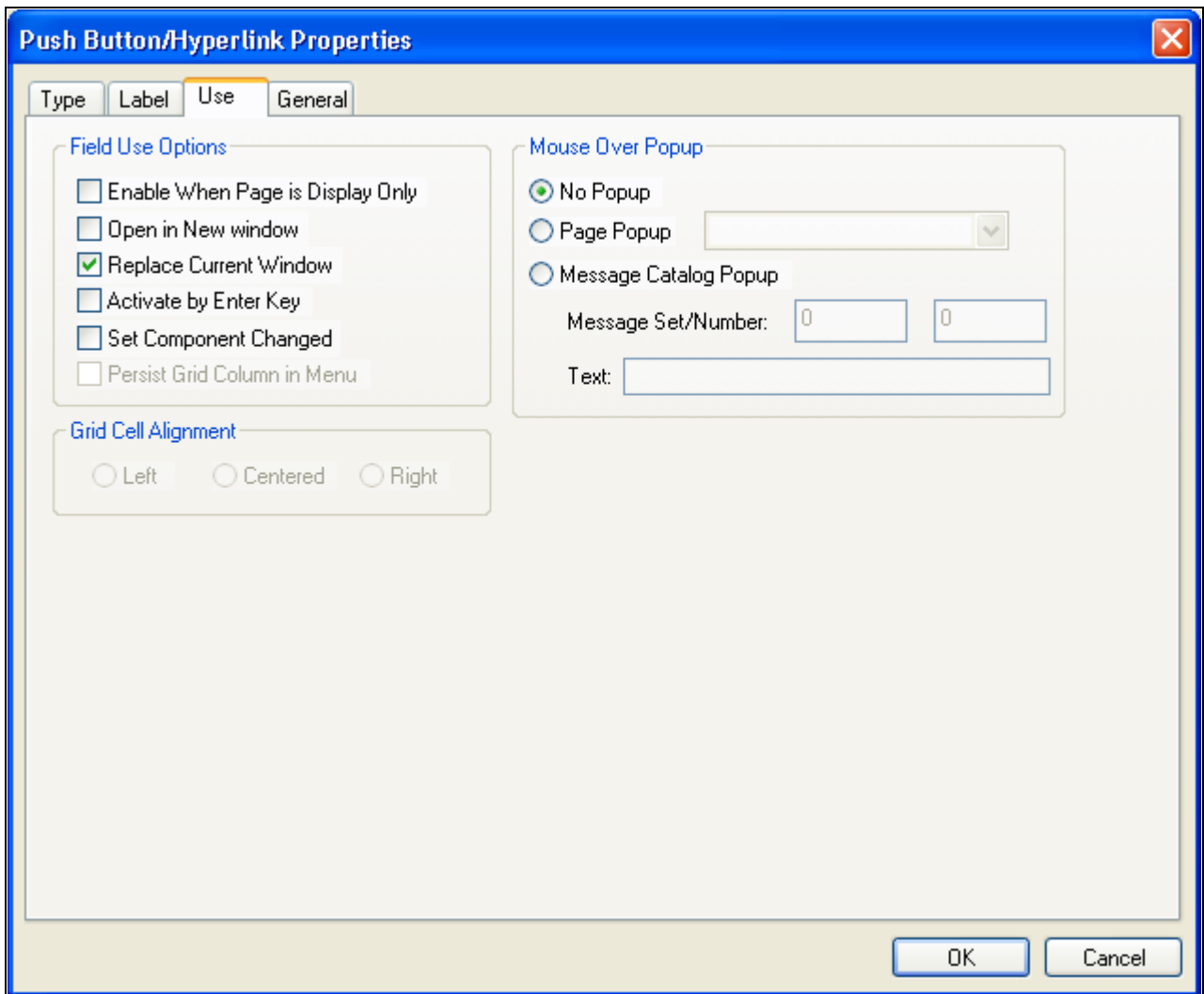
7. (Optional) Select the image that will appear on a mouse over.

For example, you might want to highlight an image so that users can easily see which option they are selecting.

8. (Optional) Select an image that will appear when the button is disabled.

## Specifying Use Properties for Push Buttons or Links

Access the Push Button/Hyperlink Properties dialog box and click the Use tab.



Push Button/Hyperlink Properties dialog box: Use tab

### **Field Use Options**

This table describes the Field Use options:

<b>Enable When Page is Display Only</b>	Select to make the push button or link available to users who have display-only access to this page.  Display-only access is set in the permission list properties.
<b>Open in New Window</b>	Select to open the target content page in a new browser window. This option is enabled when the destination type is either <i>Internal Link</i> or <i>External Link</i> . This option and <i>Replace Current Window</i> are mutually exclusive.
<b>Replace Current Window</b>	Select to replace the existing content in the target area. This option is enabled when the destination type is either <i>Internal Link</i> or <i>External Link</i> . This option and <i>Open in New window</i> are mutually exclusive.

**Activate by Enter Key**

Select to activate the push button or link when the user presses the Enter key. You can have more than one push button or link with this feature set per page, as long as only one of these fields is enabled at runtime. Otherwise, only one push button or link with this attribute enabled is allowed per page. This feature is available for all control types except external links.

When using Activate by Enter Key in a scroll area or grid with an occurs count greater than one, the Enter key triggers the first row only.

Note that on a secondary page with the standard OK and Cancel buttons enabled, the Enter key activates the OK button rather than a custom button or link.

**Set Component Changed**

Enables the system to issue a save warning for the field if the user changes the value and attempts to exit the page before saving. This setting reinforces current save warning behavior. This option is set by default for edit boxes, long edit boxes, radio buttons, check boxes, and drop-down list boxes. Deselecting this option overrides the save warning that normally appears to users. The user will not see a save warning after changing the value of the field and attempting to exit the page. An example of when this might be necessary is for a drop-down list box that includes links to another page. If the user selects a link from this field to transfer to another page but has not made any other changes to the page, then the system is not required to issue a save warning.

By deselecting the Set Component Changed flag, you ensure that no save warning message is issued for this field. This option works for fields associated with a derived work field or database record field.

For the push button/hyperlink control, the default setting is deselected instead of selected, unlike with other page controls.

**Persist Grid Column in Menu**

Select to include the field as a list box item in the persistent search results grid. You must also select the Persist in Menu grid property for the grid that contains the field.

This property is disabled on check boxes and radio buttons.

You must enable recent search results caching at the system level in the web profile for this property to take effect.

---

**Note.** This property applies *only* to fields that appear in grids on pages that are used as custom, application-specific search pages.

---

**Grid Cell Alignment**

Push buttons with images or text are center-aligned automatically when in a grid. Otherwise, select *Left*, *Centered*, or *Right*.

---

**Note.** Grid cell alignment properties are only enabled when the control is contained within a grid and the control type is configured as a text hyperlink.

---

## Mouse Over Popup

Select the Display Only check box to enable the Mouse Over Popup options. Select from these mouse over pop-up options:

<b>No Popup</b>	Select to have no pop-up page. This value is the default.
<b>Page Popup</b>	Select to associate this control with a pop-up page that you select from the drop-down list box. Only pop-up pages appear in the Page Popup drop-down list box.
<b>Message Catalog Popup</b>	<p>Select to associate this control with a system-defined message pop-up page that displays a message from the Message Catalog. You can design and implement Message Catalog pop-up pages to be context sensitive. When you select this option, the Message Set/Number fields are enabled.</p> <p>See <i>PeopleBooks and the Online PeopleSoft Library</i>, Managing the PeopleSoft Online Library and PeopleBooks, Integrating Context Sensitivity.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Using PeopleTools Utilities," Message Catalog.</p>
<b>Message Set/Number</b>	Enter the message set and message number of the message. These fields are enabled when you select Message Catalog Popup.
<b>Text</b>	Displays the message text of the message set/number that you select. This field is display-only.

---

**Note.** Attempts to make a push button/hyperlink editable at runtime when it has a mouse over pop-up page associated with it, for example, using PeopleCode, will fail.

---

### See Also

Chapter 9, "Using Page Controls," Understanding Pop-up Pages, page 226

## Programmatically Change Hover Text for Push Buttons or Links

Hover text or mouse over text enables you to display additional or supplemental information about a button or link. Under certain circumstances, you might want to vary the hover text based on the context of application data on the page.

You can alter hover text by using the PeopleCode **HoverText** Field property. This property sets the hover text for any buttons or links associated with the field. The maximum length of hover text is 100 characters. You can use the Message Catalog to store hover text.

***Example: Changing Hover Text Programmatically***

To set hover text for a push button or link at runtime, use PeopleCode to define the `HoverText` property for a field definition.

```
QE_ABSENCE_HIST.QE_ABSENCE_TYPE.HoverText = MsgGetText(95, 5037,  
"Personalize Layout");
```

---

**Note.** The `HoverText` Field class property is available for the push button/hyperlink page control only.

---

## Chapter 10

# Using Scroll Areas, Scroll Bars, and Grids

This chapter provides an overview of scroll areas, scroll bars, and grids and discusses how to:

- Use scroll areas and scroll bars.
- Use grids.
- Configure embedded help for scroll areas and grids.

---

## Understanding Scroll Areas, Scroll Bars, and Grids

A page must reflect the underlying table structures so that the system knows where to store data in the database. When you have more than one underlying record definition on a page, the role of scroll areas, scroll bars, and grids in page processing is important. Scroll controls and grids define parent and child record definition relationships on a page.

You assign an occurs level to each scroll control or grid on the page to indicate the relationship between the record definitions and the controls and to determine how the data is processed. The primary record on a page at level 0 has no scroll area or occurs level associated with it. If the page contains a record that is subordinate to the primary table, it has a scroll control with an occurs level of 1. A table that is subordinate to the level 2 record has a scroll area or grid with an occurs level of 2. PeopleTools does not support nesting beyond three levels.

PeopleSoft Application Designer automatically nests the scroll area after you set the occurs levels sequentially. Each field that you place on the page after each scroll area is automatically placed inside the scroll area preceding it until you place the next level-based control on the page.

---

## Using Scroll Areas and Scroll Bars

This section provides an overview of scroll areas and scroll bars and discusses how to:

- Set scroll area general attributes.
- Set scroll area label properties.
- Set scroll area use properties.
- Set scroll bar label properties.
- Set scroll bar use properties.

- Control scrollable data.
- Convert scroll bars to grids.

## Understanding Scroll Areas and Scroll Bars

Scroll areas, rather than scroll bars, are the preferred control for representing multiple rows of data from a table because they are easier to use during design time and offer a wider array of features. The final product at runtime appears more contained because the default setting places a border around the data.

The benefits of using scroll areas rather than scroll bars are:

- The navigation tools that you select for a scroll area are automatically positioned on the navigation bar or footer bar.

For scroll bars, you must place these buttons and links on the page manually.

- Scroll areas provide the option of showing or hiding a border around the data.
- Scroll areas provide the option of a View All button and a Find feature.
- Scroll areas enable you to determine the text or image that appears for the action items in the navigation bars.
- Scroll areas provide a row separator when you select multiple occurrences of data.

### ***Scrolls, Record Relationships, and Page Processing***

A page must reflect the underlying table structures so that the system knows where to store data in the database. When you have more than one underlying record definition on a page, the role of scroll areas and scroll bars in page processing is important. Scroll controls define parent and child record definition relationships on a page.

You assign an occurs level to each scroll control on the page to indicate the relationship between the record definitions and the controls and to determine how the data is processed. The primary record on a page at level 0 has no scroll area or occurs level associated with it. If the page contains a record that is subordinate to the primary table, it has a scroll area with an occurs level of 1. A table that is subordinate to the level 2 record has a scroll area with an occurs level of 2. PeopleTools does not support nesting beyond three levels.

PeopleSoft Application Designer automatically nests the scroll area after you set the occurs levels sequentially. Each field that you place on the page after each scroll area is automatically placed inside the scroll area preceding it until you place the next level-based control on the page.

### ***Scroll Areas***

A scroll area looks like a group box with various controls. The title bar in a scroll area, however, can also serve as a navigation bar. The navigation bar gives the user various options for viewing the data. The plus and minus buttons in the body of the scroll enable the user to add and delete rows of data.



The screenshot shows a 'Messages' window with a title bar containing 'Find | View 100 First 1 of 695 Last'. Below the title bar, it displays 'Last Update Timestamp: 06/04/1999 8:24AM'. The main area contains the following fields:

- \*Message Number: 1
- \*Severity: Error
- \*Message Text: Invalid use of !.
- Description: The exclamation symbol (!) can only be used as a token within PeopleCode as part of a not-equal operator (!=).  
Either make the operator a proper not-equal expression or delete the

The 'Description' field is a scroll area, indicated by the vertical scrollbar on its right side.

Example of a scroll area

### See Also

Chapter 11, "Creating Page Definitions," Level-Based Controls, page 314

## Setting Scroll Area General Attributes

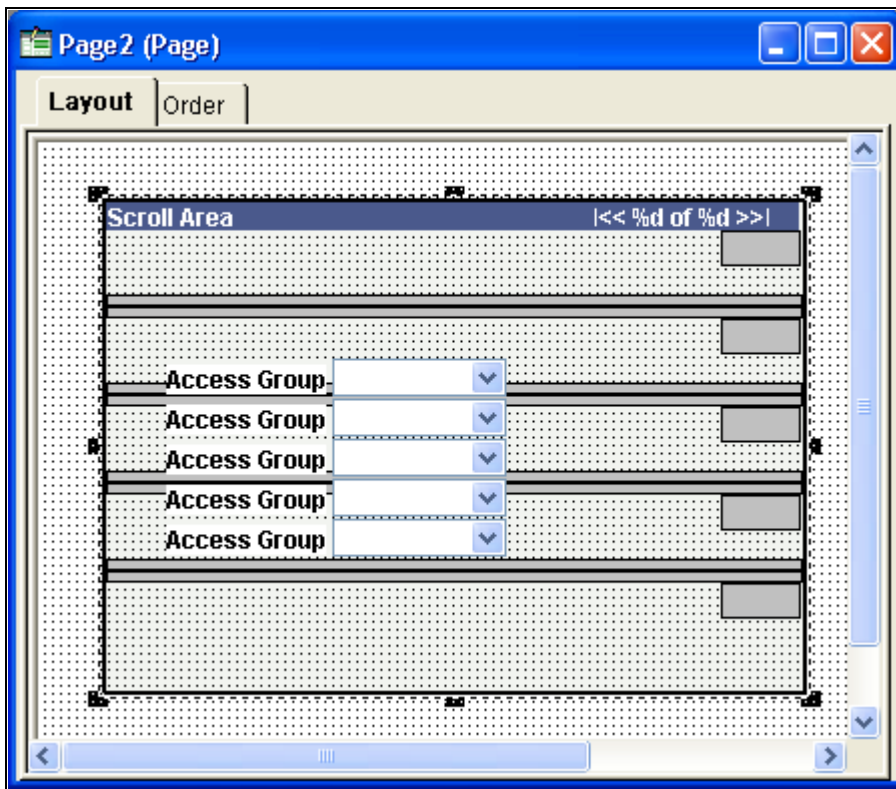
Assign the general attributes of a scroll area. These settings are the same for a scroll area as they are for a grid control.

See [Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Setting Grid General Properties, page 290.](#)

### *Manipulating Fields After Setting the Occurs Count*

If you set the occurs count to a number greater than 1, you must reposition the fields in the scroll area. If you change the occurs count after placing fields in the scroll area, they might appear one on top of the other, very close together. Unless you indicate otherwise (in the body area of the Label tab), row separator lines might appear to distinguish one row from the next.

In the following example, the occurs count was set to 5 after placing one field, Access Group, in the scroll area. You must reposition the fields and possibly the labels so that they are evenly spaced in the row separator lines.



Multiple occurs counts in a scroll area

Before you do so, it is helpful to know a few rules about how you can move these fields in relationship to one another. For example, you do not need to move individual fields to their proper destination. For this example, we call the first field in the column of fields, or the original field that you placed in the scroll area, the head field. The fields following the head field are other occurrences.

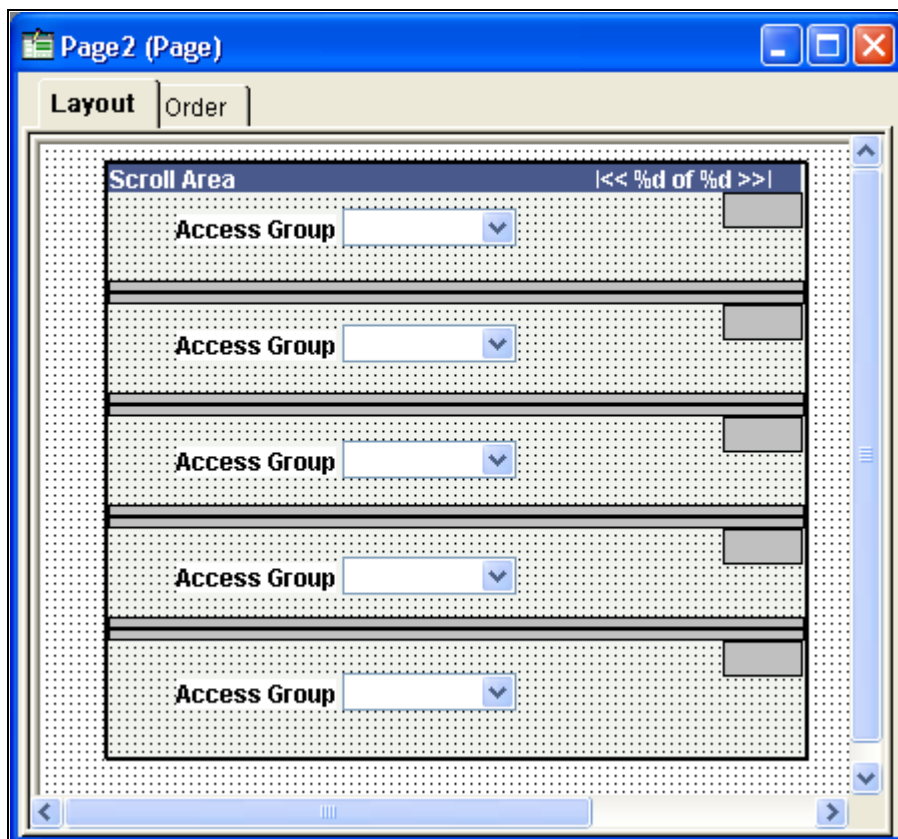
- The head field controls the horizontal movement of all other occurrences that are below it.
- You cannot move the other occurrences horizontally.
- To evenly space the occurrences vertically, drag the last occurrence in the column to the appropriate position between the bottom edge of the scroll area and the last row separator.

All occurrences above it reposition themselves equidistant from the others.

- You cannot move the add and delete push buttons (represented by the gray boxes on the right side of the scroll area) horizontally on the page.

You can move them only vertically in the same manner as the fields.

After you move the lowest field, the scroll area should look similar to this:



Fields moved to accommodate multiple occurs counts

## Setting Scroll Area Label Properties

You can configure links or push buttons in three places on a scroll area to help users navigate through multiple rows of data in a scroll area. These options and the associated settings are the same as those for grids. This section references the setting definitions for grid controls that are described later in this chapter.

### *Header Area Properties*

Use the header area properties to configure the title properties, navigation bar properties, and embedded help properties associated with the scroll area. These properties enable users to access controls for viewing the previous, next, top, or bottom row; to control the settings for the Find, Download, and View All features; and to access embedded help.

The Display Header check box is selected by default and enables access to Header Area properties. To disable the header, deselect the Display Header check box.

To configure title properties for the scroll area:

1. Select the Display Title check box.
2. Select the Left, Center, or Right title alignment option.
3. Click the Display Title Properties button to open the Title Properties dialog box.

4. Enter the Label Text that you want to appear in the scroll area title.

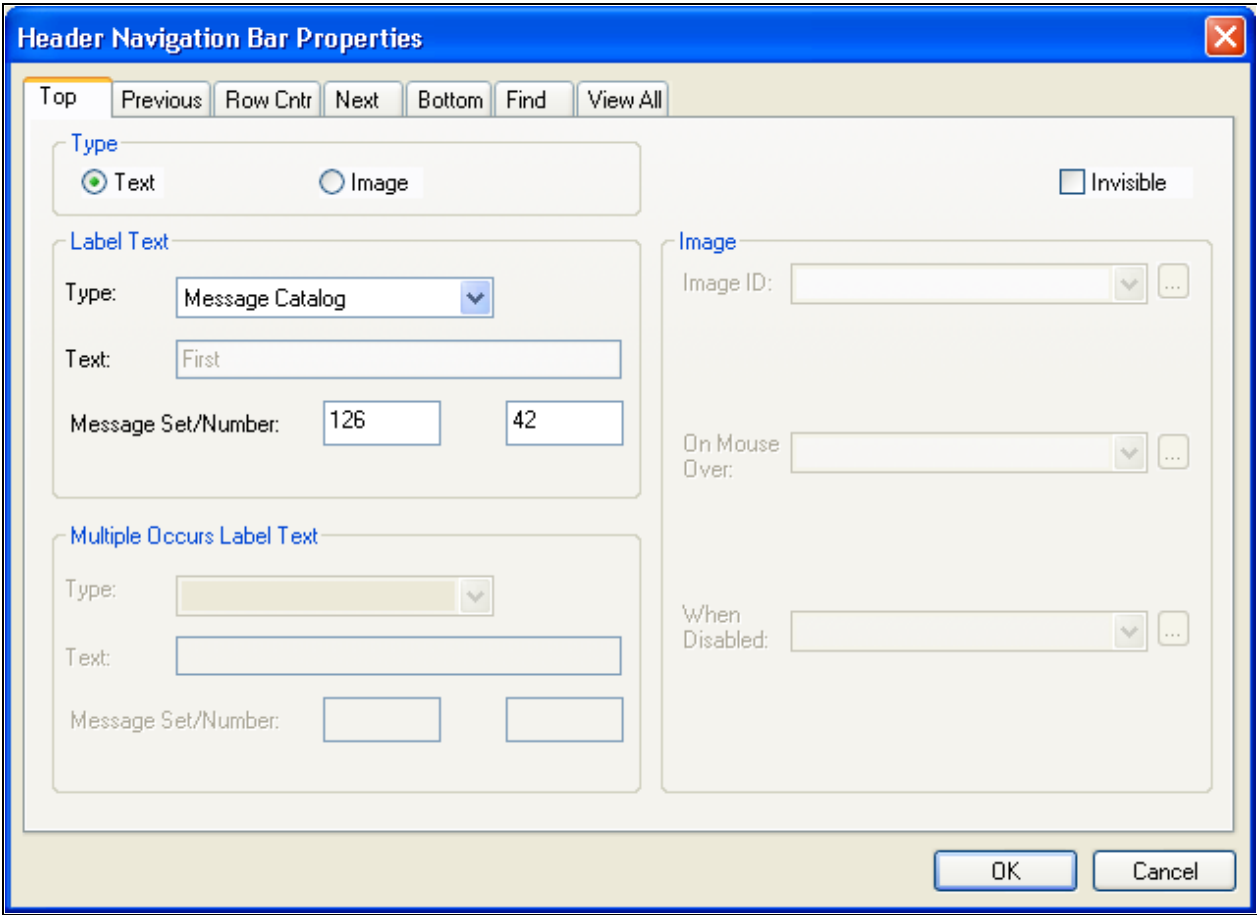
<b>Type</b>	Select the type of label text to appear in the navigation bar. <i>Message Catalog</i> : Select this option to use a message in the Message Catalog as the grid title. <i>Static</i> : Select this option to enter manual text as the grid title.
<b>Text</b>	Enter the text to appear as the title of the scroll area. This field is enabled when you select <i>Static</i> as the Label Text type.
<b>Message Set/Number</b>	Enter the message set number and message number of the message in the Message Catalog to appear as the title of the scroll area. This field is enabled when you select <i>Message Catalog</i> as the Label Text type.

To configure navigation bar properties for the scroll area:

1. Select the Display Navigation Bar check box.
2. Select the Left, Center, or Right navigation bar alignment option.

3. Click the Display Navigation Bar Properties button to open the Header Navigation Bar Properties dialog box.

The Header Navigation Bar Properties dialog box contains seven tabs, each pertaining to different settings on the navigation bar. Each tab has the same settings. All settings are set to display by default. Select the Invisible check box that appears in the upper right corner of each tab to hide the tab attribute from the user.



Header Navigation Bar Properties dialog box

This table describes the tabs in the Header Navigation Bar Properties dialog box:

<b>Top</b>	Controls the attributes for how a user returns focus to the top of the grid or the first row of data in the grid. <i>Text</i> is the default setting using <i>First</i> from the Message Catalog. You can also enter static text.
<b>Previous</b>	Controls how a user can move to a previous row in the grid. <i>Image</i> is the default setting, and the Show Previous Row button is the default image.

<b>Row Cntr</b> (row counter)	Tracks the number of rows in the grid and in which row the cursor focuses. <i>Text</i> is the default setting, and, for example, <i>1–6 of 6</i> appears in the navigation bar if the grid has six rows.  The Image Type is not enabled on this tab.
	<hr/> <b>Note.</b> The row count that appears is controlled by the occurs count that you set on the General tab in the Grid Properties dialog box (unless the user displays all). Using the previous example, if the occurs count is set to 5, then the row counter is <i>1–5 of 6</i> . <hr/>
<b>Next</b>	Focuses on the next row in the grid. <i>Image</i> is the default setting, and the Show Next Row button appears to the right of the row counter.
<b>Bottom</b>	Controls the attributes for how a user moves focus to the last row of the grid. <i>Text</i> is the default setting using <i>Last</i> from the Message Catalog. You can also enter static text.
<b>Find</b>	Displays a link that enables the user to search any field in the grid or scroll area. You can select either text or an image to represent the Find feature. The Find feature searches for matches in edit boxes, display-only fields, disabled fields, dynamic links, and text in long edit boxes. It does not search column headings, field labels, text values in icons, static links, and hidden fields, nor does it search outside the grid.
<b>View All</b>	Enables the user to view all rows of data at once. <i>Text</i> is the default setting using <i>View All</i> as the Message Catalog entry.

To configure embedded help pop-up properties for the scroll area, use these options:

<b>No Popup</b>	Select this option to disable embedded pop-up help.  This value is the default.
<b>Message Catalog Popup</b>	Select this option to enable the embedded pop-up help icon on the grid at runtime. Selecting this option enables the Message Set/Number edit boxes.
<b>Message Set/Number</b>	Enter the message set number and message number of the message catalog entry to appear when the user clicks the embedded help icon.
<b>Text</b>	Displays the message that appears when the user clicks the embedded help icon. This field is display-only.

See *PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Using PeopleSoft Application Pages," Accessing Embedded Help.

### View All Feature

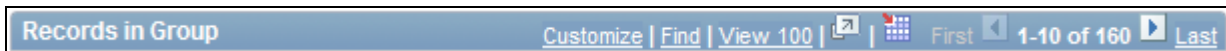
The View All feature provides a threshold to enable the user to view the data in manageable pieces. If the database returns a number of rows for the grid or scroll area that is greater than the threshold, the View All link displays "View <threshold value>." This feature also modifies the occurs count and the values for the number of non-displayed rows left to scroll through.

The Next and Previous buttons retrieve rows according to the size of the threshold. If the database returns a total number of rows for the grid or scroll area that is less than the View All threshold, then the View All link on the navigation bar remains and functions as is.

The following example uses the View All threshold, occurs count, and total rows in database values:

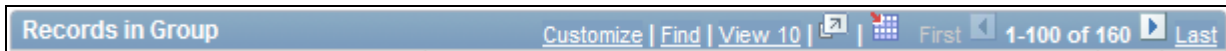
- View All threshold = 100
- Occurs count = 10
- Total rows in database = 160

The result is:



Viewing 10 rows

When the user selects View 100, the view value changes to reflect the original occurs count set for the grid. The value for the number of non-displayed rows changes to 100 to match the View All threshold, and the View 100 display changes to View 10:



Viewing 100 rows

If the Occurs Count, which you define in the properties for grid and scroll areas, is set to a number greater than or equal to the view all threshold, then the threshold takes precedence. The View All display changes to View <threshold value> and the link is unavailable. If you select Unlimited Occurs Count, the grid or scroll area displays all rows in the buffer.

### Body Area Properties

Use the body area properties to configure the row action buttons and set display properties for the body of the scroll area.

This table describes the Body Area properties:

#### Row Action Buttons

Select to display Insert and Delete buttons on each grid row at runtime. The action buttons can appear on the left or right side of the scroll area. Click the Properties button to open the Body Area Row Action Button Properties dialog box.

**Background Style**

Select a background style to display for the scroll area as a whole. The style selected here applies to all areas of the scroll area that are not covered by other elements.

**Show Border**

Select to display a standard border around your scroll area. The default is selected.

**Show Row Separator**

Select to display vertical separator lines between rows when multiple rows are returned from the database. The default is selected.

**Adjust Layout for Hidden Fields**

Select to have the scroll area adjust its borders to accommodate any hidden fields.

---

**Note.** For the scroll area to adjust automatically, the hidden fields inside the scroll area must be surrounded by open space or have other hidden fields to the right or left.

---

Click the Row Action Buttons Properties button in the Body Area group box to open the Body Area Row Action Button Properties dialog box. It contains two tabs: Insert and Delete. The default settings for these tabs are to display the Insert and the Delete buttons. Alternatively, you can select static or message catalog text, or you can make these items invisible so that they do not appear.

Body Area Row Action Button Properties dialog box



**Footer Area Properties**

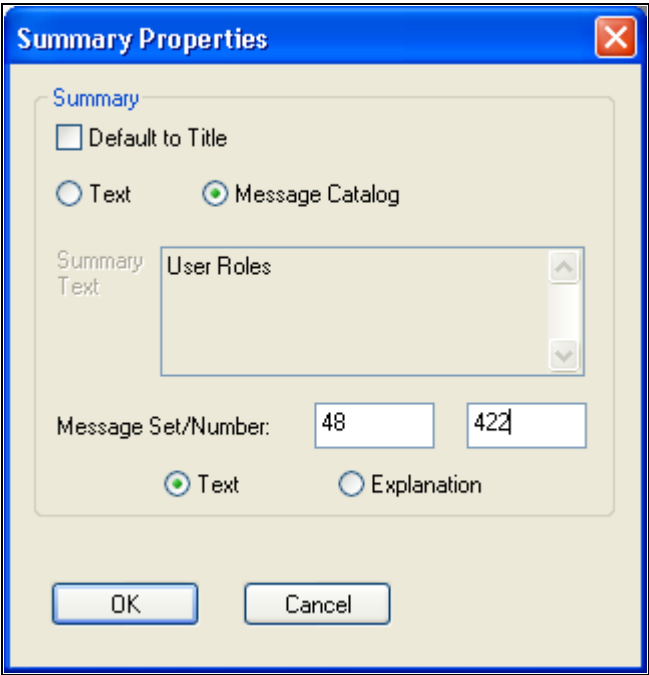
In the footer of your scroll area, you can display row action buttons and a navigation bar. To enable either of these features, select the appropriate check box, Display Row Action Buttons or Display Navigation Bar, and click the Properties button to configure the settings. Clicking the Properties button opens a separate dialog box: Footer Row Action Button Properties or Footer Navigation Bar Properties respectively.

The row action buttons are the same for the footer area as those described for the body area. The navigation bar property options are the same as those described for the header area. However, all other navigation bar settings for the footer are invisible by default.

**Summary Properties**

Summary properties enable you to provide a brief description of the functionality and content of the scroll area. This property is pertinent for users who access the application by using screen readers. The default value for the Summary Text field is the same as the Title of the scroll area. You can also enter static text or use the Message Catalog to store the summary information.

Access the Summary Properties dialog box.



Summary Properties dialog box

This table describes the summary properties of the grid:

<b>Default to Title</b>	Select this check box to have the summary property the same as the grid title. Clear this check box to activate the Text and Message Catalog options.
<b>Text</b>	Select this option to enter up to 254 characters in the Summary Text field. Selecting this option disables all Message Catalog option related fields.

**Message Catalog**

Select this option to choose a message stored in the Message Catalog. Selecting this option disables all Text option related fields. Select one of these two options:

- *Text* - Select this option to use only the message text from the message catalog.
- *Explanation* - Select this option to use only the message explanation from the message catalog.

To change the Summary properties:

1. Click the Properties button located in the Summary group box.
2. Clear the Default to Title check box to activate the other Summary options.
3. Enter static text or enter a Message Set and Number to retrieve information from the message catalog.
4. Click the OK button.
5. Save the page.

**See Also**

Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Setting Grid Label Properties, page 291

**Setting Scroll Area Use Properties**

You can select several options to determine how you use the scroll area. As with the label properties, the settings on the Use tab of the Scroll Area Properties dialog box are the same as those on the Use tab of the Grid Properties dialog box, except that the scroll area contains fewer attributes than a grid.

When working with the row counter when designing pages in PeopleSoft Application Designer, consider the following:

- If the header area navigation bar is set to invisible, the system displays no header area navigation bar row counter button. Likewise, if the footer area navigation bar is set to invisible, the system displays no footer area navigation bar row counter button. If the row counter button of the header area navigation bar is set to invisible, the system displays no header area navigation bar row counter button.
- If the row counter button of the footer area navigation bar is set to invisible, no footer area navigation bar row counter button is displayed. Otherwise, if not set to invisible, the row counter button will be displayed as "|<< your\_label >>|" where your\_label is the specified text.
- If the text of the label is an empty string or the label refers to the message catalog with a message set of 0 and a message number of 0, then "your label" appears as an empty string and the label will be rendered as "|<< >>|".

**See Also**

Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Setting Grid Use Properties, page 301

## Setting Scroll Bar Label Properties

To help identify scroll bars on the page definition, document the purpose of each scroll bar with an informational field label.

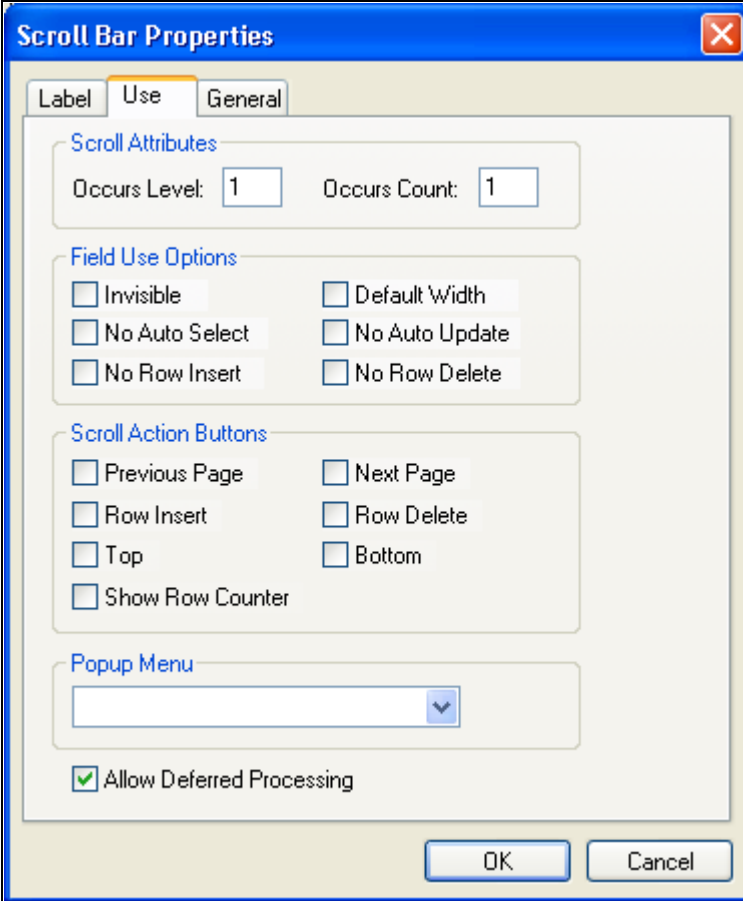
To specify a scroll bar label:

1. Access the Scroll Bar Properties dialog box.
2. Select the Label tab.
3. Enter the text label in the Text field.

This label is for your information only—it doesn't appear on the page. This label is useful for reordering page controls in the order list if you include a meaningful identifier in the label, such as the primary record definition for the scroll bar. For example, you might want to label it *ABSENCE\_HIST\_SCROLL*.

## Setting Scroll Bar Use Properties

You set the scroll action buttons on the Use tab of the Scroll Bar Properties dialog box:



The screenshot shows the "Scroll Bar Properties" dialog box with the "Use" tab selected. The dialog has three tabs: "Label", "Use", and "General". The "Use" tab contains the following sections:

- Scroll Attributes:**
  - Occurs Level:
  - Occurs Count:
- Field Use Options:**
  - ☐ Invisible
  - ☐ No Auto Select
  - ☐ No Row Insert
  - ☐ Default Width
  - ☐ No Auto Update
  - ☐ No Row Delete
- Scroll Action Buttons:**
  - ☐ Previous Page
  - ☐ Row Insert
  - ☐ Top
  - ☐ Show Row Counter
  - ☐ Next Page
  - ☐ Row Delete
  - ☐ Bottom
- Popup Menu:**
  - A dropdown menu with a downward arrow.
- ☒ Allow Deferred Processing

At the bottom of the dialog are "OK" and "Cancel" buttons.

Specifying scroll bar Use

Access the Scroll Bar Properties dialog box. Select the Use tab.

### ***Scroll Attributes***

#### **Occurs Level**

Having an occurs level for the scroll bar enables you to designate the hierarchical parent and child relationship. Entering 1 specifies that the scroll bar is at the first level. Entering 2 specifies that a scroll is at the second level and is a child of, or nested in, the first level of data, and so on up to 3. A scroll can be nested in another scroll or scroll area, but it cannot be nested in a grid.

See [Chapter 11, "Creating Page Definitions," Level-Based Controls, page 314.](#)

#### **Occurs Count**

Enter the number of rows to appear at one time in the grid at runtime.

See [Chapter 11, "Creating Page Definitions," Level-Based Controls, page 314.](#)

---

**Note.** Always reorder your controls, if necessary, before you enter an occurs count.

---

### ***Field Use Options***

#### **Invisible**

Select to make the scroll bar invisible.

#### **Default Width**

This does not apply because the scroll bar does not appear on the page at runtime.

#### **No Auto Select**

Select to prevent the system from retrieving data automatically from the database. Select to populate the scroll bar with data using the ScrollSelect PeopleCode functions.

#### **No Auto Update**

Select to prevent the system from updating data automatically based on the existing key list. Select when the scroll bar contains work field controls that should affect only a page and not the underlying database.

#### **No Row Insert**

Select to prevent the row insert function so that the user cannot insert new rows.

#### **No Row Delete**

Select to prevent the delete function so that the user cannot delete rows.

### ***Scroll Action Buttons***

If you select a Scroll Action Button option, it appears on the page next to the scroll bar in PeopleSoft Application Designer. You must manually move it to the appropriate location on the page definition.

<b>Previous Page</b>	Select to display a link that enables the user to move to the previous row or set of rows in the scroll buffer.
<b>Row Insert</b>	Select to display a button that enables the user to add a row.
<b>Top</b>	Select to display a link that enables the user to go to the first row of the scroll.
<b>Show Row Counter</b>	Select to display a counter to show users what rows they are viewing and the total number of rows in the scroll bar, for example: <1 of 3>.
<b>Next Page</b>	Select to display a link that enables the user to move to the next row or set of rows in the scroll buffer.
<b>Row Delete</b>	Select to display a button that enables the user to delete a row.
<b>Bottom</b>	Select to display a link that enables the user to go to the last row of the scroll bar.

## Controlling Scrollable Data

You can control scrollable data using the PeopleCode Data Buffer Access Classes.

See *PeopleTools 8.52: PeopleCode Developer's Guide*, "Accessing the Data Buffer."

## Converting Scroll Bars to Grids

You can convert a single-level scroll bar to a grid control. PeopleSoft Application Designer provides a utility to automate this task. The Convert Scroll to Grid function:

- Designates the size and position of the grid as set to the area covered by the left-most label to the scroll bar.
- Issues warnings before deleting controls that are not supported by the grid.
- Converts radio buttons to drop-down list boxes.
- Sets label alignment of check boxes to center.

The conversion utility provides warnings for scroll bars that can't be converted to grids due to grid control limitations. The conversion utility validates to the following limitations:

- No more than one grid can exist on a page.
- The scroll bar to be converted must not have any nested scroll areas or bars under it.
- The grid control must be the last control on the page.
- Radio buttons are not supported in grids and must be replaced with drop-down list boxes.

To convert a scroll to a grid:

1. Select the scroll bar.
2. Right-click and select Convert Scroll To Grid from the pop-up menu.

This menu option is available only for scroll bars that can be converted. This utility searches for all page fields that are defined in the field order after the selected scroll bar control, up until the next scroll bar or scroll area control. It does the following for each field:

- Confirms that the scroll control type can be displayed in a grid control.
  - Creates a grid column of the appropriate type.
  - Populates the attributes.
3. Click OK to display the new grid.

If any of the page fields cannot be displayed in a grid or if other problems are found, the utility displays an error message explaining why the scroll bar cannot be converted to a grid.

---

## Using Grids

This section provides overviews of grids and discusses how to:

- Insert and resize grid controls.
- Insert and manipulate grid columns.
- Set column properties.
- Create a tabbed grid.
- Freeze grid columns.
- Use multiple grids on a page.
- Enable grid personalization.
- Set grid general properties.
- Set grid label properties.
- Set grid use properties.

---

**Note.** A specific type of grid called an analytic grid can be used exclusively for applications built for the PeopleSoft Analytic Calculation Engine.

---

### See Also

*PeopleTools 8.52: Analytic Calculation Engine, "Creating Analytic Grids"*

## Understanding Grids

Scroll controls and grid controls function very similarly. Grids are more compact and display fields in a spreadsheet-like format. You insert page controls into a grid and associate the controls to record definitions, just as you do in a scroll control.

You can insert these page controls into a grid:

- Check boxes.
- Drop-down list boxes.
- Edit boxes.
- Long edit boxes.
- Push buttons or links.
- Images.
- HTML areas.
- Secondary pages.
- Subpages.
- Tab separators.

---

**Note.** When designing grids, use prompts in place of drop-down list boxes for better performance.

---

## Understanding Grid Properties

When you accept the default settings of grids, the grid appears with :

- A data navigation bar at the top of the table that enables the user to page through additional rows of data.
- Delete and Add buttons at the end of each row that enable the user to insert and delete rows in the grid.

The inserted row appears under the current row. These buttons appear automatically as determined by the grid properties (display-only, no row insert, and no row delete).

- An occurs count of *1*.

You can set the grid to be any size at design time and then set the occurs count to control the maximum number of rows to display at runtime.

There are three areas on a grid to which you can apply labels or image buttons to help the user navigate through the data in the grid. Set properties for these areas:

- Header
- Body
- Footer

## Understanding Grid Personalization

PeopleSoft Application Designer enables the user to personalize a grid at runtime. By default, grid personalization is enabled. To change the default settings for grid personalization, open the Grid Properties dialog box and click the Display Navigation Bar Properties button. Grid personalization settings appear on the Personalize tab in the Header Navigation Bar Properties dialog box.

**Note.** If you do not display the grid header bar, the Personalize link is not available to the user unless you enable it for the footer area.

**Personalize Column and Sort Order**

To order columns or add fields to sort order, highlight column name, then press the appropriate button.  
Frozen columns display under every tab.

**Column Order**

- Permission List
- Description
- View Definition

☐ Hidden  
☐ Frozen

**Sort Order**

- Permission List

☐ Descending

OK Cancel Preview Copy Settings

Permission List	Description	View Definition
HCCPAM1	Absence Mngmnt Administrator	View Definition
HCCPAMSS01	Absence Self-Service	View Definition
HCCPSCAW1030	AWE Employee	View Definition
HCCSPWEBLIB	Standard WebLib Permissions	View Definition

Personalize Column and Sort Order page

When the user clicks the Personalize link for a grid on a transaction page, the system opens the Personalize Column and Sort Order page, where custom parameters can be set for that grid. This page contains a sample grid that displays a few rows of real data from the buffer in display-only format. If there is no data in the buffer for that grid, only the column headings appear. Tabs in the grid are active so that the user can view all sample data.

From this personalization page, users can control the column and sort order, as well as copy other users' grid settings or choose to share their settings with other users once they save their changes.

The Column Order grid displays the columns in their current order, including tab separators. The list of fields is driven by the page definition but excludes columns that are hidden by PeopleCode. The user can hide columns and change their sort order. Users can also freeze grid columns. Columns that you freeze at design time automatically appear to the user as already frozen. However, the user has the option of overriding this setting.



The Sort Order grid enables the user to control the ascending and descending sort of each column, with the exception of tab separators, images, push buttons, links, and HTML areas. Users can sort hidden fields, however. The default setting is ascending for each column.

After making changes, the user can accept, preview, restore defaults, or cancel selections. If the user saves the new settings by selecting OK, the system stores the new settings as persistent values for that user based on a key structure. Therefore, each time the user accesses this grid, the system uses all personalization options that were set previously. The user can restore all default settings by selecting the Delete Settings link to remove all stored personalization settings from the database.

The user can override the persistent sort order directly on the transaction page by clicking the linked field label to affect the sort. Clicking the link repeatedly toggles between ascending and descending sort order for that column. This type of sorting directly in the transaction page is *not* persistent.

---

**Note.** You can continue to implement the SortScrolls PeopleCode command, but the user can override the defined sort with this feature.

---

---

**Note.** Modifying a grid or a page definition that contains a grid removes all runtime user personalizations associated with the modified page and the grid.

---

## Inserting and Resizing Grid Controls

To insert a grid on a page:

1. Select Insert, Grid.
2. To adjust the grid width, drag the horizontal or vertical control handles.

The horizontal grid width should be roughly equivalent to the columns that you insert into the grid. Otherwise, your grid might appear wider than necessary at runtime.

## Inserting and Manipulating Grid Columns

This section discusses how to:

- Insert grid columns.
- Delete grid columns.
- Move grid columns on the Layout tab.
- Move grid columns on the Order tab.
- Resize grid columns.

### ***Inserting Grid Columns***

To insert a grid column:

1. Use one of these methods to insert a column:

- Select a page control from the Insert menu or the page control toolbar and click the grid.
- Drag a page field from inside the current page or from another page.
- Copy and paste a page field or record field.
- Drag a definition (such as a record field, a page field, or an entire record definition) from the project workspace.

The new location of the definition or control is determined by the location of the upper left-hand corner of the drag rectangle, or by the mouse cursor if no drag rectangle appears. PeopleSoft Application Designer automatically places the first column on the left-hand side of the grid.

2. Click your mouse or release the field in the gray area of the grid.

Place new columns to the right of existing columns by clicking or releasing the control in the gray area of the grid where no columns are present. To place a column between two grid columns, release the new column in the first of the two columns.

When inserting subpages into your grid as a column, be aware that:

- You can insert subpages by dragging them from the project workspace or by cutting and pasting from the page to the grid.
- Only one column displays in the grid in PeopleSoft Application Designer representing all fields in the subpage.

At runtime, all fields display in the order they appear on the Order tab of the subpage.

- You cannot insert subpages that contain fields that are not allowed in grids such as static text or radio buttons.

The system generates an error indicating the invalid field type. Similarly, if at a later point an invalid field type is added to a subpage already in a grid, the system generates a runtime error when the user accesses the page containing the subpage.

- The subpage you insert can contain fields that are associated with subrecord record fields.

The same record name substitution mechanism that applies to subpages inserted onto a page applies to those inserted as grid columns. The system replaces the name of the subrecord as part of the expansion process and it is never seen at runtime.

See [Chapter 9, "Using Page Controls," Understanding Subpages, page 237](#).

---

**Note.** When inserting rich text enabled long edit boxes in grids, be aware that the long edit boxes must be display only. If the Display Only property is not enabled, unexpected results will occur.

---



---

**Note.** For scrollable grids that contain an HTML area column, the height of the grid is the product of the number of rows x 28 pixel default height per row; the grid height is not based on the data in the HTML area.

---

### ***Deleting Grid Columns***

To delete a grid column:

1. Select a column by right-clicking the column heading.

Be sure that you select only the column and not the grid as a whole; otherwise, you might delete the entire grid instead of just the column. The grid is selected when control boxes appear around the edges of the grid. The column is selected when it turns black.

2. Select Delete from the pop-up menu.

### ***Moving Grid Columns on the Layout Tab***

To move grid columns on the Layout tab:

1. Select a column by clicking the column heading.
2. Drag the column to its new location.
3. Release the mouse button over the column that is to the left of the new location.

### ***Moving Grid Columns on the Order Tab***

To move grid columns on the Order tab:

1. Select the column row by clicking the row number.

The grid and grid columns are clearly distinguished from other page fields on your page. The grid is identified by the Type column. All columns in the grid are directly below in a lighter green. Nongrid page field rows are white.

2. Drag the row to the new position in the grid.

A red line indicates the new position of the column before you release it. You cannot move a column outside of the grid when working on the Order tab. Similarly, you cannot move an existing page field from elsewhere on the page into the grid. Both of these operations can be performed on the Layout tab.

### ***Resizing Grid Columns***

You can resize grid columns in PeopleSoft Application Designer by dragging the right border of the column heading.

## **Setting Column Properties**

After you insert the page control or field into your grid, you can set the properties for that field as you would set properties for any other page control. Access the field properties by double-clicking the column heading.

Column headings and their alignment are derived from the label text that you select on the Label tab of the page field properties dialog box. The alignment of column contents is derived from the alignment settings on the Record tab of the page field properties dialog box. If alignment settings aren't available on the Record tab, then the system uses the Label alignment settings. For column headings, use the following guidelines for label alignment. Do not use colons in column labels.

<b><i>Field Type</i></b>	<b><i>Label Alignment</i></b>
Push button	Center Align
Link	Left Align
Character	Left Align
Numeric	Right Align
Check box	Center Align
Field data	Auto (automatic)

### ***Default Image Field Dimensions in Scrollable Grids***

When working with scrollable layout grids that include image fields, consider these default behaviors:

- The default row height for 28pix.
- The header label width determines the default column width.
- Grid cell dimensions do not resize automatically to accommodate a stored image with dimensions that exceed the default dimension.

If you have images with dimensions that exceed the default dimensions of (28 x (*column width*)) then users are unable to see the entire image because the grid cell is smaller than the image. To prevent images from being cropped by the page control:

1. Determine and record the dimensions of the largest image in the field.
2. Access the Image properties dialog box of the image page field.
3. On the Record tab, enter the recorded values from step one in the Width and Height fields.
4. Click the OK button and save the page definition.

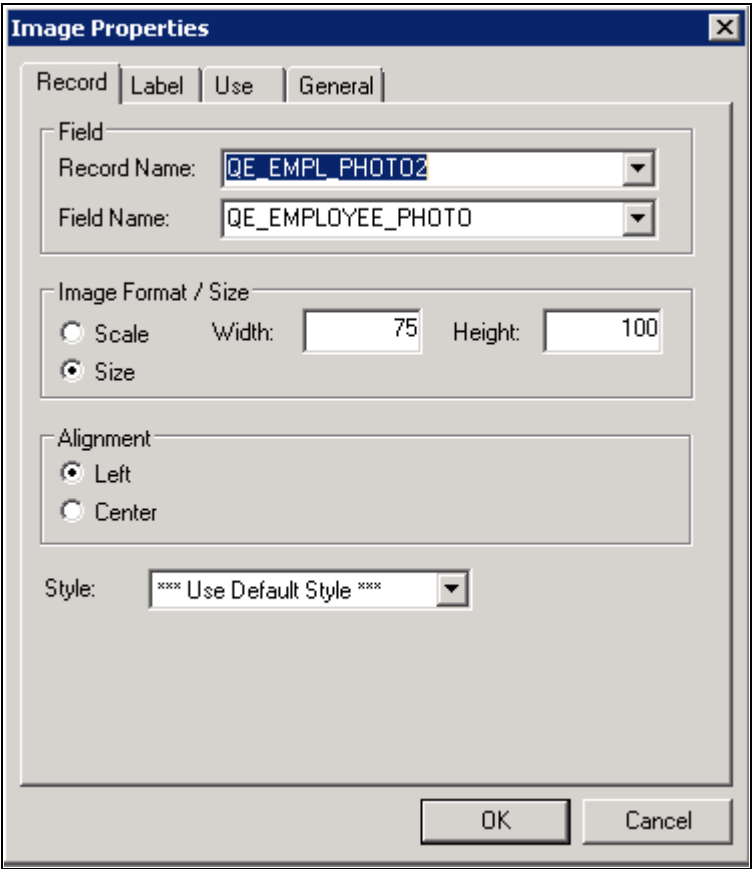


Image Properties dialog box with non-default size values

Creating Tabbed Grids

Oracle recommends using a tabbed grid design if your grid contains many columns that extend past the right side of the page. Do this by adding tab separator controls to your grid.

Define Personalizations						
Option Category Level: PeopleTools						
Define Personalizations						
Customize   Find   View All   1-25 of 48   First   Last						
*User Option	Description	*Option Category Group	Option Category	User Option Type	Locale Based	
ACCESS	Accessibility Features	PS Internet Architecture	General Options	System	<input type="checkbox"/>	+ -
ACEGRDCOLS	Max Col/View All-Analytic Grid	PS Internet Architecture	Navigation Personalizations	System	<input type="checkbox"/>	+ -
ACEGRDROWS	Max Row/View All-Analytic Grid	PS Internet Architecture	Navigation Personalizations	System	<input type="checkbox"/>	+ -
ADBTN	Tab over Add/Del Buttons (+/-)	PS Internet Architecture	Navigation Personalizations	System	<input type="checkbox"/>	+ -
ADES	Afternoon designator (PM, pm)	PS Internet Architecture	Regional Settings	System	<input checked="" type="checkbox"/>	+ -

Tabbed grid

You can give users the option of expanding all of the columns to the right so that they are visible when using the browser's horizontal scroll bar. This is particularly useful to power users who do a great deal of intensive data entry. The Enable View All Columns check box on the Use tab in the Grid Properties dialog box controls this feature and is selected by default. The Expand All button appears to the right of the grid tabs, as shown in the preceding screen shot. After the grid is expanded, the Show Tabs button appears to enable the user to collapse the grid so that the tabs appear again.

---

**Note.** Each time the user moves to a new tab results in a transmission to the web server to display the next page.

---

To create a tabbed grid:

1. Insert a grid control on the page.
2. Insert the columns to appear on the first tab.
3. Insert a tab separator control after the last column.
4. Set the tab separator properties.
  - a. Double-click the tab separator to access the Tab Separator Properties dialog box.
  - b. Specify the label properties on the Label tab.
  - c. Specify the general properties on the General tab.

5. Insert the next set of columns to appear on the second tab.

Alternatively, you can place all of the columns in your grid first and then insert the tab separator between the two columns that you want separated.

6. Repeat steps 3 and 4 if necessary.
7. Set the tab separator properties by double-clicking the column heading of the tab separator.
  - a. On the Label tab, specify the label type.

Select *None*, *Text*, or *Message Catalog*. If you select *Text*, enter the label text and select the label alignment. The label text that you select appears on the tab in the grid.

- b. On the General tab, determine whether you want to enable the tab separator as a page anchor.

If so, select the Enable as Page Anchor check box and enter a page field name.

8. Save the page.

---

**Note.** The grid tab must contain at least one non-frozen and visible column or the tab does not appear.

---

## Freezing Grid Columns

You can freeze the first column or the first several columns in a tabbed grid so that they appear on subsequent tabs. To freeze more than one column, select the freeze option for each column that you want to appear on the following tab.

Freezing grid columns impacts the horizontal scrolling ability of the grid at design time. If you freeze a grid column that is not visible inside the boundaries of the grid at design time, you are not able to view all columns in your grid. Lengthen the width of your grid to accommodate all columns.

Freeze a grid column by selecting the Freeze Grid Column check box on the Use tab of the properties dialog box for that column. Note that the effects of selecting this attribute are visible only at runtime.

## Using Multiple Grids on a Page

You can place as many grids on a page as you want, provided that they are at the same occurs level. They can be one above the other or they can be side-by-side, such as those in the following example. This is helpful when you must transfer data from one grid to another without switching between pages.

In the following example, the user can transfer data from the Source Competencies grid on the right to the Assigned Competencies grid on the left by using the double left arrow button. The push button is associated with a PeopleCode program, enabling it to transfer the data between the two grids.

The screenshot displays a web application interface with a top navigation bar containing four tabs: **Competency Evaluation**, **Competency Assignment**, **Competency Rating**, and **Competency Verification**. Below the tabs, the user information is shown: **Smith, John** (Employee), **ID: FG7025**, and a [Link to Career Planning](#).

The main content area is divided into two side-by-side grids. The left grid is titled **Competency Assignment** and has a sub-header **Assigned Competencies**. It includes a **Populate from Job** button and a table with the following data:

Description	Category
Ability to listen & respond	Ability
Set & achieve goals	Ability
Debates issues unabrasively	Ability
Conceptual thinking	Ability
Abstract thinking	Ability

The right grid is titled **Source Competencies** and has a sub-header **Source Competencies**. It includes a **Search by:** section with **Category** (Ability) and **Type** (Cust Servc) dropdowns. Below this is a table with the following data:

Description
<input type="checkbox"/> Service Orientation
<input type="checkbox"/> Relationship Building
<input type="checkbox"/> Develop & implement solutions

A double left arrow button (←←) is positioned between the two grids, indicating a transfer function. At the bottom of the interface, there are several buttons: **Save**, **Return to Search**, **Next in List**, **Previous in List**, **Update/Display**, **Include History**, and **Correct History**. A navigation bar at the very bottom contains links: [Competency Evaluation](#) | [Competency Assignment](#) | [Competency Rating](#) | [Competency Verification](#).

Grids side-by-side on a page

As you can with other level-based controls, you can nest a grid in a scroll area or scroll bar. However, you cannot nest a grid in another grid.

## Enabling and Disabling Grid Personalization

Grid personalizations are enabled by default. To disable grid personalization:

1. Open the Grid Properties dialog box.
2. Select the Label tab.

3. Click the Personalize button for the header navigation bar.

This opens the Header Navigation Bar Properties dialog box.

4. Select the Personalize tab.
5. Select the Invisible check box.
6. Click the OK button.

To enable grid personalization in the navigation footer:

1. Open the Grid Properties dialog box.
2. Select the Label tab.
3. Select the Display Navigation Bar check box in the Footer Area group box.
4. Click the Properties button for the footer navigation bar.

This opens the Footer Navigation Bar Properties dialog box.

5. Select the Personalize tab.
6. Clear the Invisible check box.
7. Click the OK button.

## Setting Grid General Properties

Access the Grid Properties dialog box. Select the General tab.

### Main Record

Select the record that contains most of the fields that you want to use in the grid. Fields that you display in the grid that are not from the main record should be display-only or related fields, which you can set on the Use tab of the properties dialog box for that column.

### Page Field Name

Enter a name for the grid. The default is the main record for the grid, however, you can rename the grid as long as it is a *unique* name for the page or component. This name is used by the PeopleCode GetGrid function to create a grid definition.

See *PeopleTools 8.52: PeopleCode API Reference*, "File Class" and *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," GetGrid.



<b>Occurs Level</b>	<p>Enter a 1, 2, or 3 to represent at which level the grid resides on the page. Having an occurs level for the grid enables you to designate the hierarchical parent-child relationship. Entering 1 specifies that the grid is at the first level. Entering 2 specifies that a grid is at the second level and is a child of, or nested in, the first level of data, and so on up to 3. A grid can be nested in another level-based control, but it cannot be nested in another grid.</p> <p>See <a href="#">Chapter 11, "Creating Page Definitions," Level-Based Controls, page 314.</a></p>
<b>Occurs Count</b>	<p>Enter the number of rows to display initially at runtime.</p> <p>See <a href="#">Chapter 11, "Creating Page Definitions," Multiple Occurrences of Data, page 319.</a></p>
<b>Unlimited Occurs Count</b>	<p>Select instead of setting an occurs count if you want the system to display all rows of data in the buffer for this grid. Navigation links do not appear at runtime and the size of the grid depends on the number of rows in the buffer.</p>
<b>Enable as Page Anchor</b>	<p>Select to apply an anchor tag to serve as a jump destination on the page.</p>

## Setting Grid Label Properties

Access the Grid Properties dialog box. Select the Label tab.

**Grid Properties**

General **Label** Use

☒ Display Header

**Header Area**

☒ Display Title ☒ Left ☐ Center ☐ Right [Properties](#)

☒ Display Navigation Bar ☐ Left ☐ Center ☒ Right [Properties](#)

Style: [Use Default Style](#)

**Embedded Popup Help**

☐ No Popup

☒ Message Catalog Popup

Message Set/Number:

Text:

**Body Area**

Row Action Buttons ☐ Left ☒ Right [Properties](#)

Background Style: [Use Default Style](#)

☒ Show Border ☐ Show Grid Lines ☒ Show Row Headings (runtime)

**Column Headings**

☒ Show Column Headings (runtime) Style: [PAGRIDHEADERTXT](#)

☐ Display Footer

**Footer Area**

☐ Display Row Action Buttons ☒ Left ☐ Center ☐ Right [Properties](#)

☐ Display Navigation Bar ☐ Left ☐ Center ☒ Right [Properties](#)

Style: [Use Default Style](#)

**Summary**

Summary attribute for Accessibility [Properties](#)

[OK](#) [Cancel](#)

Grid Properties dialog box: Label tab

The Grid Properties, Label tab provides five main grid setting areas for you to configure:

- Header Area.
- Body Area.
- Column Headings.
- Footer Area.

- Summary.

Certain grid label properties such as label text and summary text can also be set through PeopleCode.

See *PeopleTools 8.52: PeopleCode API Reference*, "Grid Classes," Grid Class Properties.

### **Header Area Properties**

Use the header area properties to configure the title properties, the navigation bar properties, and embedded help properties associated with the grid. These properties enable users to access controls for viewing the previous, next, top, or bottom row; to control the settings for the Find, Personalize, Download, and View All features; and to access embedded help.

The Display Header check box is selected by default and enables access to Header Area properties. To disable the header, clear the Display Header check box.

To configure title properties for the grid:

1. Select the Display Title check box.
2. Select Left, Center, or Right title alignment option.
3. Click the Display Title Properties button to open the Title Properties dialog box.
4. Enter the label text that you want to appear in the grid title.

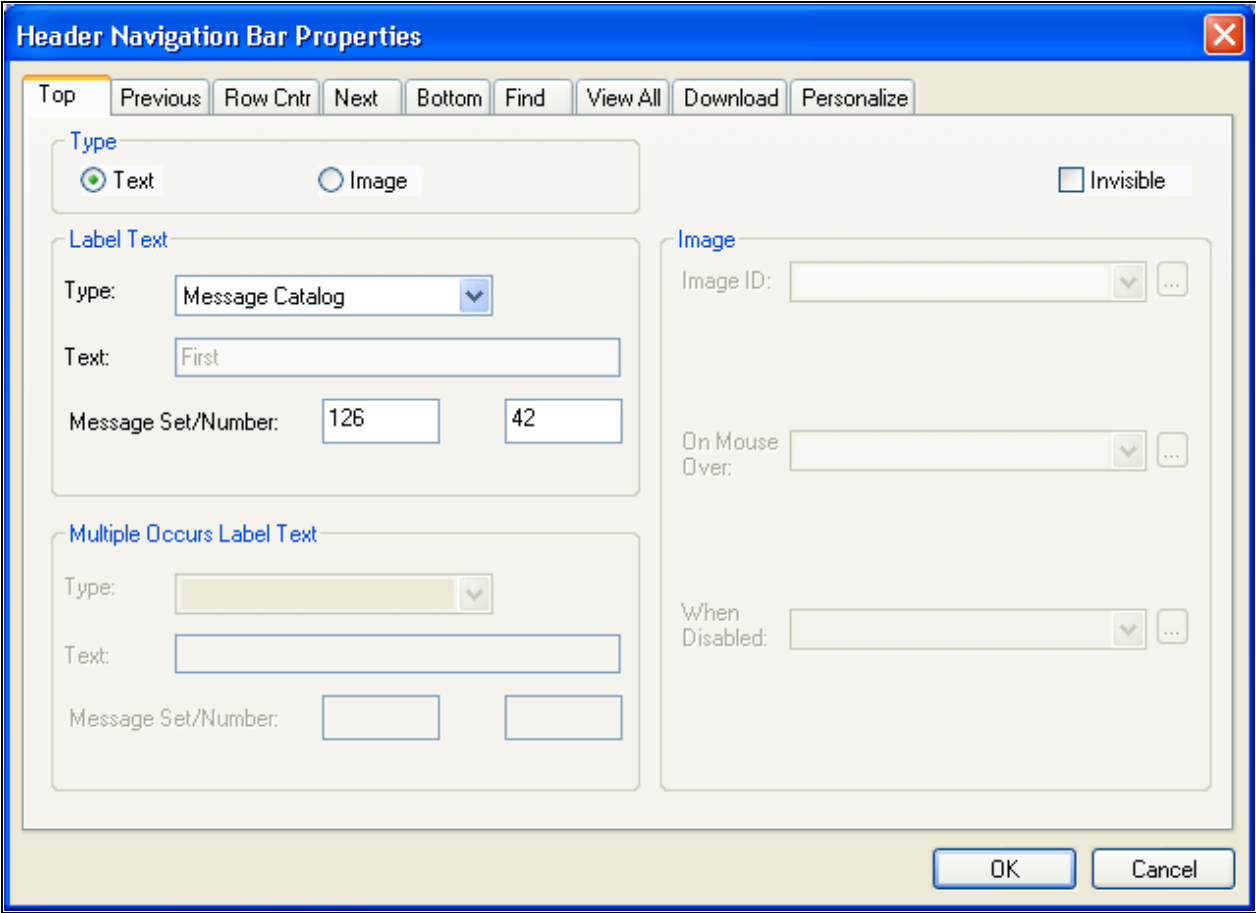
<b>Type</b>	Select the type of label text to appear in the navigation bar. <i>Message Catalog:</i> Select this option to use a message in the Message Catalog as the grid title. <i>Static:</i> Select this option to enter manual text as the grid title.
<b>Text</b>	Enter the text to appear as the title of the grid. This field is enabled when you select <i>Static</i> as the Label Text Type.
<b>Message Set/Number</b>	Enter the message set number and message number of the message in the Message Catalog to appear as the title of the grid. This field is enabled when you select <i>Message Catalog</i> as the Label Text Type.

To configure navigation bar properties for the grid:

1. Select the Display Navigation Bar check box.
2. Select Left, Center, or Right navigation bar alignment option.

3. Click the Display Navigation Bar Properties button to open the Header Navigation Bar Properties dialog box.

The Header Navigation Bar Properties dialog box contains seven tabs, each pertaining to different settings on the navigation bar. Each of these tabs has the same settings. All settings are set to display by default. Select the Invisible check box that appears in the upper right corner of each tab to hide the tab attribute from the user.



Header Navigation Bar Properties dialog box

This table describes the tabs on the Header Navigation Bar Properties dialog box:

<b>Top</b>	Controls the attributes for how a user returns focus to the top of the grid or the first row of data in the grid. <i>Text</i> is the default setting, using <i>First</i> from the message catalog. You can also enter static text.
<b>Previous</b>	Controls how a user can move to a previous row in the grid. <i>Image</i> is the default setting, with the Show Previous Row button as the default image.

<b>Row Cntr</b> (row counter)	<p>Tracks the number of rows in the grid and in which row the cursor holds focus. <i>Text</i> is the default setting with, for example, <i>1-6 of 6</i> appearing in the navigation bar if there are six rows in the grid.</p> <p>The Image Type is not enabled on this tab.</p> <hr/> <p><b>Note.</b> The row count that appears is controlled by the occurs count that you set on the General tab in the Grid Properties dialog box (unless the user displays all). Using the previous example, if the occurs count is set to 5, then the row counter is <i>1-5 of 6</i>.</p> <hr/>
<b>Next</b>	<p>Brings focus to the next row in the grid. <i>Image</i> is the default setting with the Show Next Row button appearing to the right of the row counter.</p>
<b>Bottom</b>	<p>Controls the attributes for how a user moves focus to the last row of the grid. <i>Text</i> is the default setting, using <i>Last</i> from the message catalog. You can also enter static text.</p>
<b>Find</b>	<p>Displays a link that enables the user to search any field in the grid or scroll area. You can select either text or an image to represent the find feature. The Find feature searches for matches in edit boxes, display-only fields, disabled fields, dynamic links, and text in long edit boxes. It does not find column headings, field labels, text values in icons, static links, and hidden fields, nor does it search outside the grid.</p>
<b>View All</b>	<p>Enables the user to view all rows of data at once. <i>Text</i> is the default setting using <i>View All</i> as the message catalog entry.</p>

**Download**

Available for grids only. Displays a download image button in the header or footer bar that enables users to download all active, unhidden rows to a Microsoft Excel spreadsheet, regardless of how many rows are visible on the screen. The default for this setting in the header area is *Invisible*. For the footer area of a grid, the download button appears by default when you select the Display Navigation Bar check box in the Footer Area group box of the Label tab in the Grid Properties dialog box.

---

**Note.** When you download a column of links comprised of numeric values, the values are downloaded as numbers instead of as quoted strings unless doing so causes Excel to modify the format of the numbers.

---

**Note.** Microsoft Excel may display Unicode comma-delimited data differently because Microsoft Excel may be expecting a tab-delimited file. If display abnormalities occur, you have two options.

In Microsoft Excel, after highlighting the first column, you can select Data, Text to Columns and select the options required to set Microsoft Excel to expect a comma-delimited file.

Or, in PeopleSoft Application Designer change the file layout from comma-delimited to tab-delimited. Keep in mind that changing the file layout properties in PeopleSoft Application Designer could cause the data to change such that other applications may no longer recognize the data. Also, changing the file layout properties may impact more than one application, if that file layout is shared.

---

**Personalize**

Available for grids only. Displays a Personalize link in the header bar that enables the user to access all grid personalization options.

See [Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Enabling and Disabling Grid Personalization, page 289](#).

To configure embedded help pop-up properties for the grid, select from these options:

**No Popup**

Select this option to disable embedded pop-up help.  
This is the default.

**Message Catalog Popup**

Select this option to enable the embedded pop-up help icon on the grid at runtime. Selecting this option enables the Message Set/Number edit boxes.

**Message Set/Number**

Enter the message set number and message number of the message catalog entry to appear when the user clicks the embedded help icon.

**Text**

Displays the message that appears when the user clicks the embedded help icon. This field is display-only.

### View All Feature

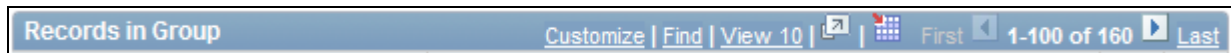
The View All feature provides a threshold to enable the user to view the data in manageable pieces. If the database returns a number of rows for the grid or scroll area that is greater than the threshold, the View All link displays "View <threshold value>." This feature also modifies the occurs count and the values for the number of non-displayed rows left to scroll through.

The Next and Previous buttons retrieve rows according to the size of the threshold. If the database returns a total number of rows for the grid or scroll area that is less than the View All threshold, then the View All link on the navigation bar remains and functions as is.

The following example uses the View All threshold, occurs count, and total rows in database values:

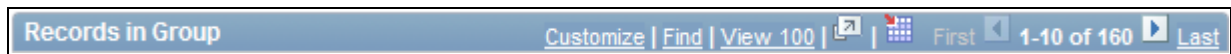
- View All threshold = 100
- Occurs count = 10
- Total rows in database = 160

These values result in the following for the user:



Viewing 10 rows

When the user selects View 100, the view value changes to reflect the original occurs count set for the grid. The value for the number of non-displayed rows changes to 100 to match the View All threshold, and the View 100 display changes to View 10:



Viewing 100 rows

If the Occurs Count, which you define in the properties for grid and scroll areas, is set to a number greater than or equal to the view all threshold, the threshold takes precedence. The View All display changes to View <threshold value> and the link is unavailable. If you select Unlimited Occurs Count, the grid or scroll area displays all rows in the buffer.

### Body Area Properties

Use the body area properties to configure the row action buttons and set display properties for the body of the grid.

This table describes the Body Area properties:

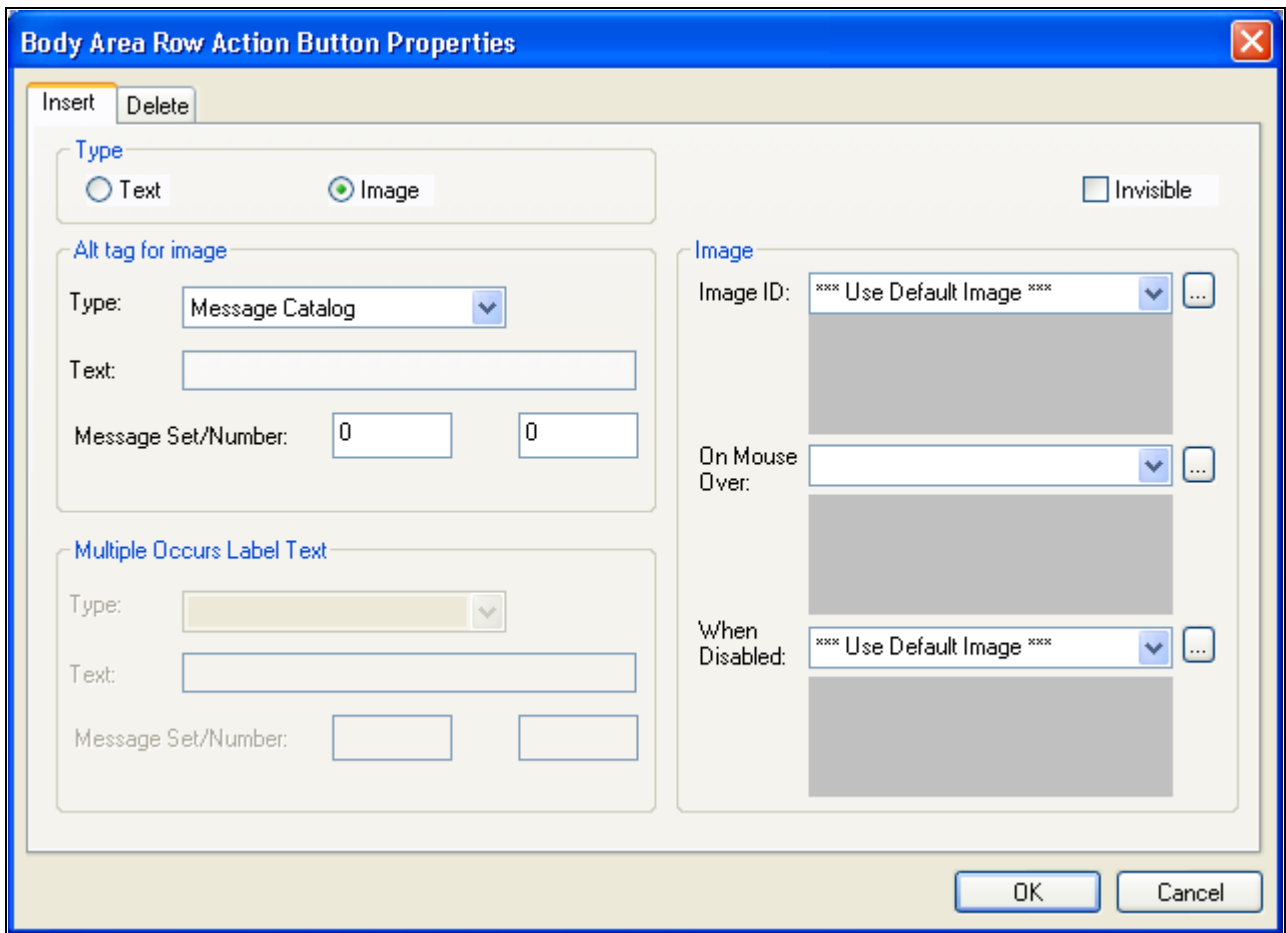
#### Row Action Buttons

Select to display Insert and Delete buttons on each grid row at runtime. The action buttons can appear on the left or right side of the grid. Click the Properties button to open the Body Area Row Action Button Properties dialog box.

<b>Background Style</b>	Select a background style to display for the grid as a whole. The style selected here applies to all areas of the grid that are not covered by other elements.
<b>Show Border</b>	Select to display a standard border around your grid. The default is selected.
<b>Show Grid Lines</b>	<p>Select to show grid lines on the grid. The default is to <i>not</i> display grid lines (this option is deselected).</p> <p>This setting should remain deselected as grid lines are set through the style class.</p>
<b>Show Row Headings (runtime)</b>	<p>Select to number the rows at runtime. This option is selected by default.</p> <hr/> <p><b>Note.</b> The row headings column appears at design time even if Show Row Headings (runtime) is not selected. This enables you to more easily select the grid if necessary.</p> <hr/>

Click the Row Action Buttons Properties button in the Body Area group box to open the Body Area Row Action Button Properties dialog box. It contains two tabs: Insert and Delete. The default settings for these tabs are to display the Insert and Delete buttons. Alternatively, you can select static or message catalog text, or you can make these items invisible so that they do not appear.





Body Area Row Action Button Properties dialog box

### ***Column Heading Properties***

You can set the grid to display column headings, and you can select the style in which to display them. Select a predefined column heading style from the drop-down list box or use the default. Changing styles alters the foreground, background, and font in the column headings, making it different from the data rows. The default for Show Column Headings is selected.

### ***Footer Area Properties***

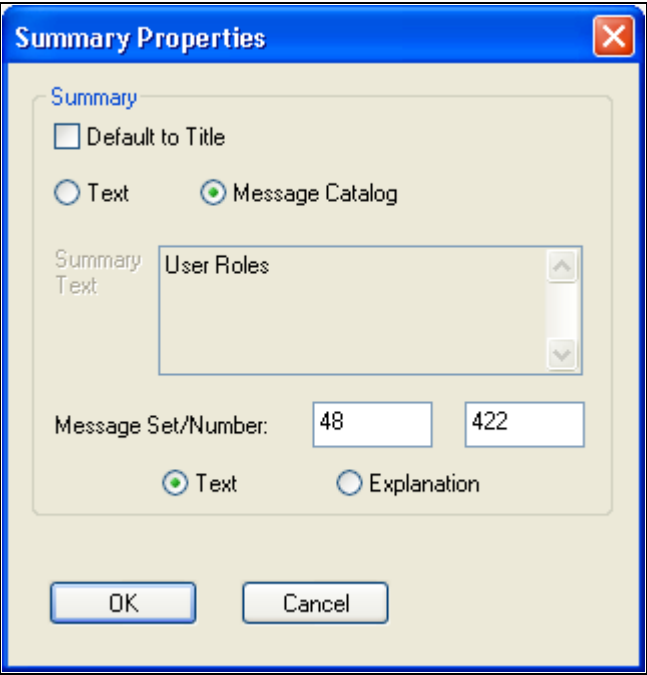
In the footer of your grid, you can display row action buttons and a navigation bar. To enable either of these features, select the appropriate check box, Display Row Action Buttons or Display Navigation Bar, and click the Properties button to configure the settings. Clicking the Properties button opens a separate dialog box: Footer Row Action Button Properties or Footer Navigation Bar Properties, respectively.

The row action buttons are the same for the footer area as those described for the body area. The navigation bar property options are the same as those described for the header area. However, all other navigation bar settings for the footer are invisible by default.

**Summary Properties**

Summary properties enable you to provide a brief description of the functionality and content of the grid. This property is pertinent for users who access the application by using screen readers. The default value for the Summary Text field is the same as the Title of the scroll area. You can also enter static text or use the Message Catalog to store the summary information.

Access the Summary Properties dialog box.



Summary Properties dialog box

This table describes the summary properties of the grid:

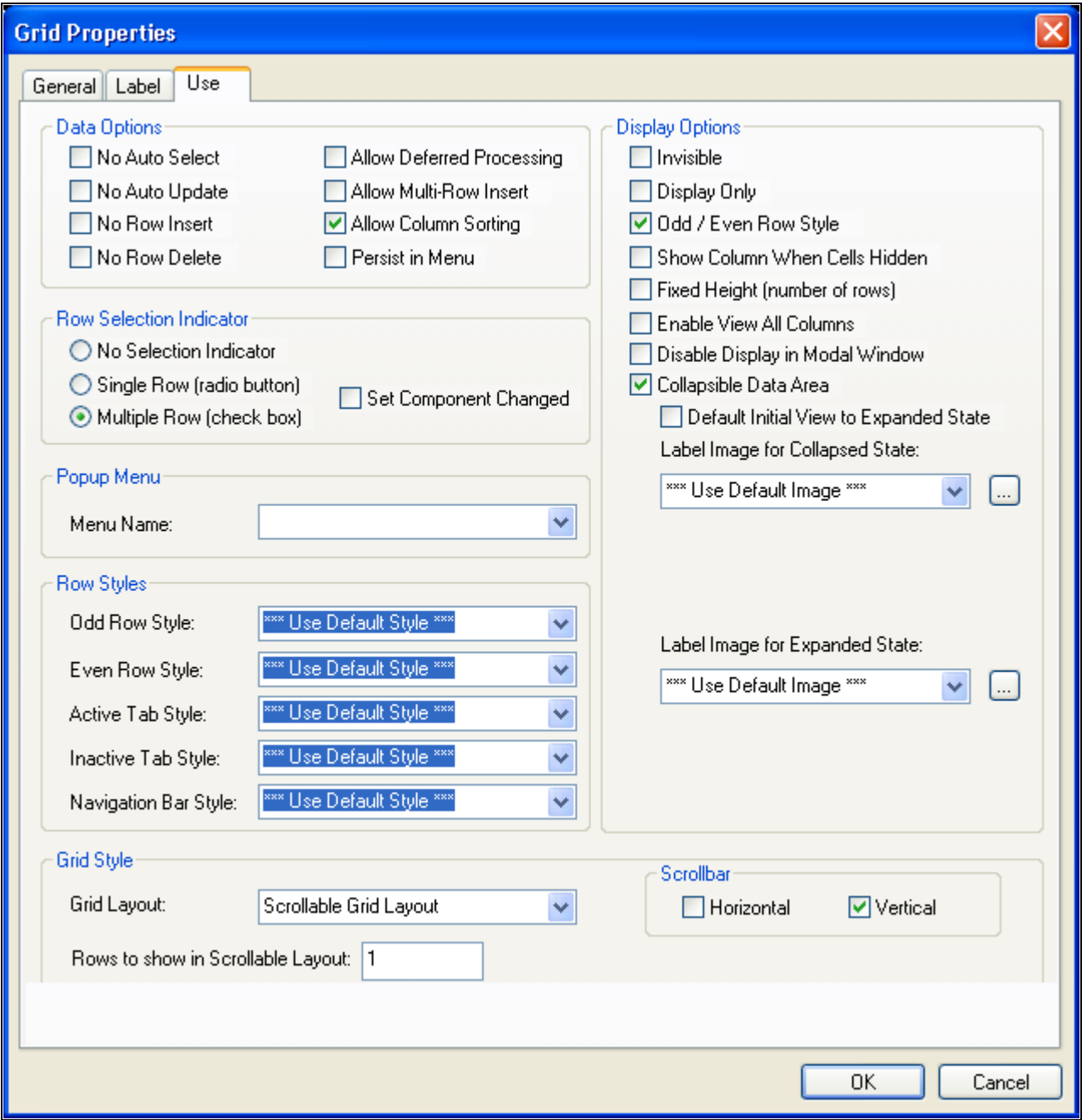
<b>Default to Title</b>	Select this check box to have the summary property the same as the grid title. Deselect this check box to activate the Text and Message Catalog options.
<b>Text</b>	Select this option to enter up to 254 characters in the Summary Text field. Selecting this option disables all Message Catalog option related fields.
<b>Message Catalog</b>	<div>Select this option to choose a message stored in the Message Catalog. Selecting this option disables all Text option related fields. Select one of these two options:</div> <ul style="list-style-type: none"><li><i>Text</i>: Select this option to use only the message text from the Message Catalog.</li><li><i>Explanation</i>: Select this option to use only the message explanation from the Message Catalog.</li></ul>

To change the Summary properties:

1. Click the Properties button located in the Summary group box.
2. Clear the Default to Title check box to activate the other Summary options.
3. Enter static text or enter a Message Set and Number to retrieve information from the message catalog.
4. Click the OK button.
5. Save the page.

## Setting Grid Use Properties

On the Use tab in the Grid Properties dialog box, you control data entry, row style, and display options, as well as whether your grid has a pop-up menu in the navigation bar at runtime. Both the grid and scroll area share many of the same attributes. Refer to the proceeding term tables for information about these shared settings.



Grid Properties dialog box: Use tab

**Data Options**

This table describes the data options properties:

<b>No Auto Select</b>	Suppresses the system from retrieving data from the database automatically. Select to populate the grid or scroll area with data using the ScrollSelect PeopleCode functions.  See <i>PeopleTools 8.52: PeopleCode Language Reference</i> , "PeopleCode Built-in Functions," RowScrollSelectNew.
<b>No Auto Update</b>	Suppresses the system from automatically updating data based on the existing key list. Select when the grid or scroll contains work field controls that should affect only a page and not the underlying database.
<b>No Row Insert</b>	Suppresses the Add button in the body area so that the user cannot insert rows. If this option is selected, PeopleSoft Application Designer automatically selects and disables the Invisible check box on the Insert tab of the properties dialog box for the body area.
<b>No Row Delete</b>	Suppresses the Delete button in the body area so that the user cannot delete rows. If this option is selected, PeopleSoft Application Designer automatically selects and disables the Invisible check box on the Delete tab of the properties dialog box for the body area.
<b>Allow Deferred Processing</b>	Set by default, this option ensures that the grid or scroll area processes in deferred mode. This setting overrides any individual field-level settings in your level-based control.
<b>Allow Multi-Row Insert (Allow Multiple Row Insert)</b>	Enables the user to add multiple blank rows at once to a grid after a page is displayed. This feature is available only in Add and Update modes. When the user clicks the Add Multiple Blank Rows button, a JavaScript prompt appears so that the user can specify the number of rows to insert; the user can insert between 1 and 99 rows.
<b>Allow Column Sorting</b>	Enables the end user to perform nonpersistent sorts in a grid at runtime by clicking the column headings. The default is selected to allow column sorting automatically.

**Persist in Menu**

Stores the most recent transaction search in browser memory or web server memory and makes the results accessible through the drop-down navigation menu structure. To specify which fields appear as columns in the search results grid, use the Persist Grid Column in Menu property.

---

**Note.** This property applies *only* to fields that appear in grids on pages that are used as custom, application-specific search pages.

---

See [Chapter 6, "Creating Record Definitions," Setting Record Field Use Properties, page 134.](#)

---

**Note.** If multiple grids exist on a custom search page and you enable this property on more than one grid, the system stores the search for only the first grid as it appears in the page order on the Order tab.

---

You must enable recent search results caching at the system level in the web profile for this property to take effect.

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Configuring the Portal Environment," Configuring Caching.

**Multirow Insert Option**

Each time that a user adds a row to a grid or scroll area requires a transmission to the application server. With the multirow insert option, this transmission occurs only once when the user wants to add multiple blank rows. Selecting this option can significantly enhance the user's experience when doing data entry and also meets the developer's goal of limiting page processing.

At design time in PeopleSoft Application Designer, this feature is disabled by default. If the No Row Insert check box is selected, then the Allow Multi-Row Insert check box is deselected and disabled. Where the multirow insert button appears, in the row or in the footer, depends on how you define the grid or scroll area properties on the Label tab.

You can set any occurs count for your grid. When the end user inserts  $n$  rows, the occurs count for the grid or scroll area is temporarily increased to  $n + 1$  (if necessary), showing one row of context plus the new rows. If you select an unlimited occurs count for the grid, this temporary adjustment is not necessary.

---

**Note.** This feature should not be enabled for grids and scroll areas with effective-dated rows.

---

When the user saves the page, normal save processing runs, including save-edits on all rows. New, unchanged rows are not saved. After save processing, new, unchanged (and unsaved) rows are deleted from the buffer. The only exception is when only one row exists and it is new and unchanged. This row remains in the buffer, even though it has not been saved, because a rowset must always contain one row.

For an end user, pressing the Alt+7 keys works the same with the multirow Add button as it does with the standard Add button.

**Row Selection Indicator**

These options enable the user to select one or more rows in the grid and scroll area. At runtime, indicators appear to the left of row numbers for grids and in the upper left corner for scroll areas. A transmission to the server triggers the selection of the row or rows that the user designates. These selections can then be detected in PeopleCode via the selected property on the row class for each row of data.

<b>No Selection Indicator</b>	This value is the default selection. Prevents the user from selecting rows on the transaction page.
<b>Single Row (radio button)</b>	Enables the user to select a single row in a grid or scroll area. Inserts a radio button to the left of each row that the user can select.
<b>Multiple Row (check box)</b>	Enables the user to select multiple rows in a grid or scroll area. Inserts a check box to the left of each row that the user can select. Users can select as many rows as needed.
<b>Set Component Changed</b>	<p>Enables the user to receive a save warning if a change is made to a row selection indicator in the grid. At runtime, the transaction page uses this option to set the PPR_CHANGED flag. This option is deselected by default.</p> <p>This property ensures that if <i>any</i> changes occur to the grid, the data is saved to the database.</p> <p>If you disable this property, any row insert, row delete, or field change on the rowset using PeopleCode would not cause the system to treat the component data as changed.</p>

---

**Note.** For the row selection options, PeopleCode can change the selected property, which affects which boxes are selected when the page is regenerated.

---

### **Popup Menu**

Select a preset pop-up menu from the drop-down list box. The pop-up menu button appears in the left-hand corner of the navigation header at runtime.

See [Chapter 13, "Creating Menu Definitions," Creating Pop-up Menus, page 430.](#)

### **Row Styles**

You can control the color, font, and other characteristics of a row, active tab, inactive tab, and navigation bar by specifying a style in the drop-down list box for each of the row style options.

### **Grid Style**

These options enable the advanced grid properties available with scrollable grid layout only.

See *PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Working With Grids and Scroll Areas," Viewing Data in Scrollable Layout Grids.

### **Grid Layout**

Select one of these settings:

- *Scrollable Grid Layout:* Sets the style for the grid to scrollable.
- *Original Grid Layout:* Sets the style for the grid to original.

**Rows to show in scrollable layout**

Enter  $n$  (number) to set the height of the scrollable grid to  $n$  visible rows.

This value represents the number of rows to show when Scrollable Grid Layout is selected and the Vertical Scrollbar check box is selected. If this number is greater than the Occurs Count set on the General tab, then no vertical scrollbar appears, regardless of whether the Vertical Scrollbar check box is selected. If this number is lower than the Occurs Count and the Vertical Scrollbar check box is selected, then a vertical scroll bar appears and this number of rows will appear in the scroll area. The scroll bar will scroll over the entire Occurs Count number of rows. To see more rows in this case, use the row navigation controls on the grids navigation bar.

If this field is set to 1, then the vertical scroll bar will not appear except in View All mode, in which case the scroll bar will display the number of rows listed in the Occurs Count setting.

The default value is 15.

**Horizontal**

Select to activate a horizontal grid scroll bar at the bottom of the scrollable grid.

---

**Note.** This option affects scrollable grid layout only.

---

**Vertical**

Select to activate a vertical grid scroll bar at the right side of the grid.

---

**Note.** This option affects scrollable grid layout only.

---

**Display Options****Invisible**

Select to make the grid or scroll area and its associated fields and records invisible. Use this setting for work grids or scrolls for which you want the underlying processing to be transparent to users. Typically, you associate invisible work grids and scrolls with PeopleCode Scroll functions that enable you to further control application processing.

**Display Only**

In some cases, you might design grids that enable users to view but not change information. Select this check box if you do not want the user to enter data into the fields in any of the rows.

**Odd/Even Row Style**

Selected by default, this attribute displays the grid with contrasting colors by row. The Odd Row Style and Even Row Style drop-down list box options appear instead of the single Row Style option in the Row Styles group box.

**Show Column When Cells Hidden**

Select if you want the columns to appear even if all of the cells in a column are hidden. Columns do not collapse.



**Fixed Height (number of rows)**

Select to set the grid to a height corresponding to the number of rows in the occurs count attribute. The grid remains fixed at that height even if the actual number of data rows varies from that number. If the fixed height property is not used, the size of the grid dynamically changes based on the occurs count and the number of data rows retrieved.

**Enable View All Columns**

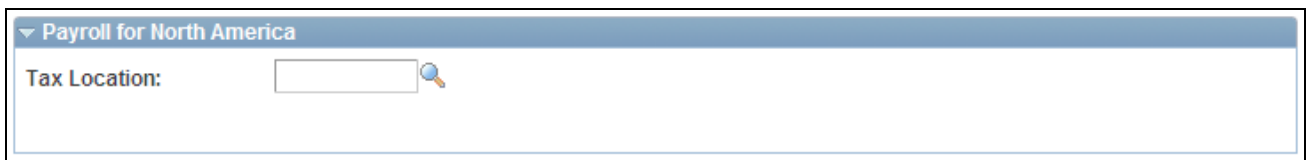
Displays an expand all button to the right of a tabbed grid, enabling the user to view all columns of the grid by scrolling to the right using the horizontal scroll bar of the browser instead of selecting the grid tabs. The Show Grid Tab button appears when columns are expanded to enable the user to view the tabs again.

**Disable Display in Modal Window**

Select to disable the zoom capability of the grid.

**Collapsible Data Area**

Select to collapse the data area for your grid or scroll area into a header bar with a small image that the user must click to activate or expand it. Selecting this option enables the Default Initial View to Expanded State check box. You can then select the label image to represent the collapsed and expanded states.



Expanded scroll area



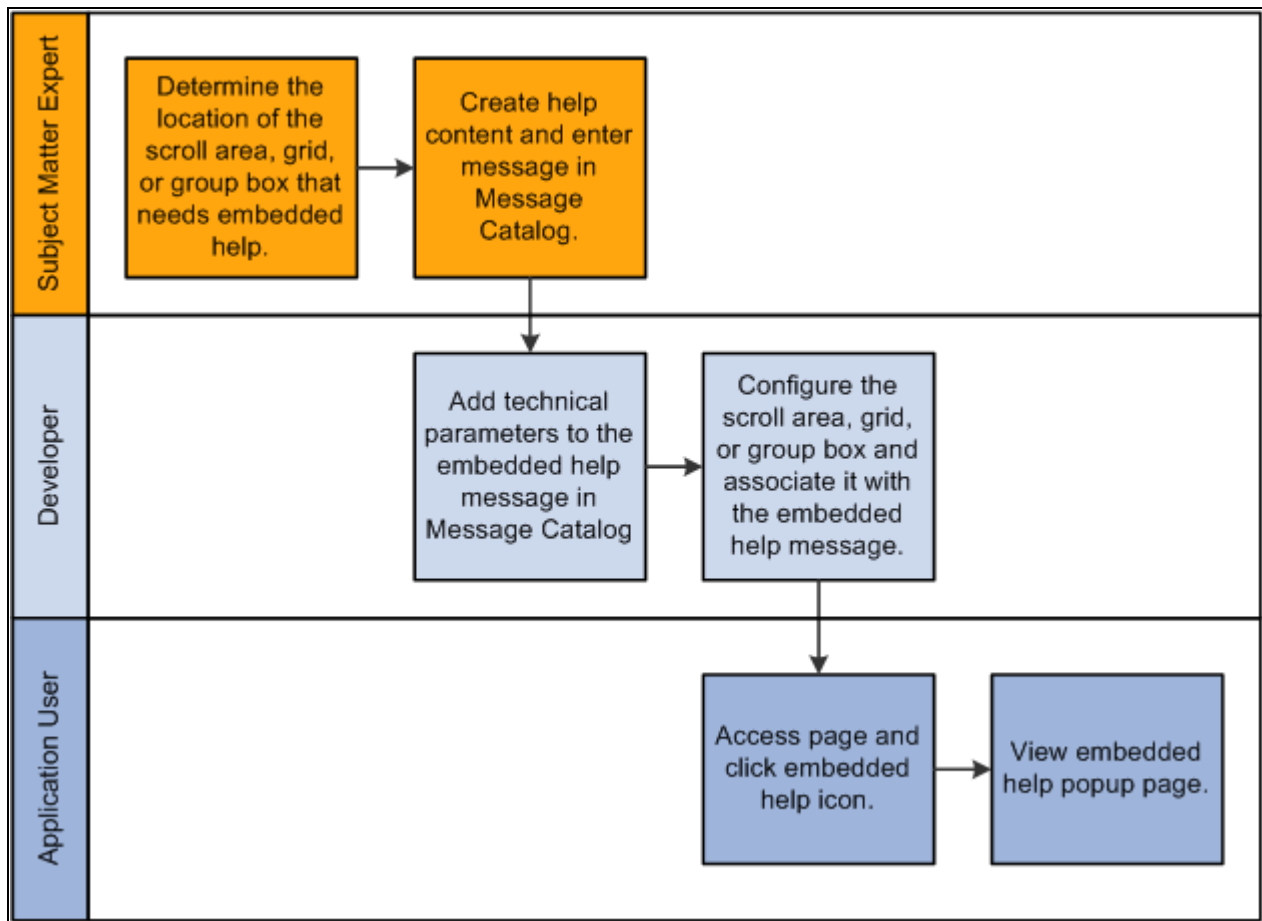
Collapsed scroll area

---

## Configuring Embedded Help for Scroll Areas and Grids

You can configure scroll areas and grids to display an embedded help pop-up message that you create and store in the Message Catalog.

This diagram illustrates the business process flow for creating, configuring, and accessing embedded help:



Business process flow for creating and accessing embedded help

This section discusses how to:

- Enter embedded help messages in the Message Catalog.
- Associate help messages with scroll areas and grids.

---

**Note.** Use this information to configure embedded help for group boxes, also.

---

## Entering Embedded Help Messages in the Message Catalog

To configure embedded help, first you must create Message Catalog messages that you specifically design for use by scroll areas and grids. To enter embedded help messages in the Message Catalog:

1. Access the Message Catalog (PeopleTools, Utilities, Administration, Message Catalog).
2. Select an existing or create a new message.

---

**Note.** You should create new message sets to avoid extra upgrade maintenance tasks.

---

3. Enter the help content in the Explanation field. Include the necessary variables if you want to generate other help links that appear on the embedded help pop-up page. The table at the end of this section describes the variables and their usage.

---

**Note.** The help content should be contextual and specific to the page control to which it will be associated in the next section.

---

4. Save the message entry.

### ***Embedded Help Variables***

This table describes the variables that you can add to the embedded help content in the Message Catalog. Use these variables to generate links on the embedded help pop-up page that enable the user to access additional online help.

---

**Note.** You must install the Online PeopleBooks Library for any PeopleBooks Online Help link to function properly.

---

<b><i>Variable</i></b>	<b><i>URL</i></b>	<b><i>Label</i></b>
<code>%PBCTXT ( )</code>	This variable uses the current page name as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so the embedded link takes the user to the PeopleBooks content associated with the current page.	Because no message is referenced, the label of the help link appears as Help.
<code>%PBCTXT ( sn , mn )</code> The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.	This variable uses the current page name as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the current page.	Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.  If you reference an invalid message, the label appears as Help.
<code>%PBCTXT ( CTXT_PG_NM )</code> The argument <code>CTXT_PG_NM</code> is a page definition name.	This variable uses the <code>CTXT_PG_NM</code> argument as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the page referenced by the argument.	Because no message is referenced, the label of the help link appears as Help.
<code>%PBCTXT ( CTXT_PG_NM , sn , mn )</code> The argument <code>CTXT_PG_NM</code> is a page definition name.  The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.	This variable uses the <code>CTXT_PG_NM</code> argument as the <code>%CONTEXT_ID%</code> to build the PeopleBooks Help URL so that the embedded link takes the user to the PeopleBooks content associated with the page referenced by the argument.	Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.  If you reference an invalid message, the label appears as Help.

<b>Variable</b>	<b>URL</b>	<b>Label</b>
<p><code>%PBURLTBL(&lt;URL_IDENTIFIER&gt;=&gt; =&gt; ,sn,mn)</code></p> <p>The argument <code>&lt;URL_IDENTIFIER&gt;</code> is any <code>URL_ID</code> that is stored in the URL Maintenance component.</p> <p>See <i>PeopleTools 8.52: System and Server Administration</i>, "Using PeopleTools Utilities," URL Maintenance.</p> <p>The argument <code>sn</code> is the Message Set Number. The argument <code>mn</code> is the Message Number.</p>	<p>This variable uses the <code>&lt;URL_IDENTIFIER&gt;</code> argument to build help URL so that the embedded link takes the user to the site associated with the <code>URL_ID</code>.</p>	<p>Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.</p> <p>If you reference an invalid message, the system uses the <code>&lt;URL_IDENTIFIER&gt;</code> argument as the label.</p>
<p><code>%PBLNK(http://www.site.com=&gt; =&gt; ,sn,mn)</code></p> <p>The argument <code>sn</code> is the Set Number. The argument <code>mn</code> is the Message Number.</p>	<p>This variable uses the hard-coded URL so that the embedded link takes the user to the specified site.</p>	<p>Because a message is referenced, the label of the help link appears as the value that is stored in the Message Text field for the referenced message.</p> <p>If you reference an invalid message, the system uses the URL as the label.</p>

### See Also

*PeopleTools 8.52: PeopleTools Portal Technologies*, "Configuring the Portal Environment," Configuring General Portal Properties

*PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," Message Catalog

## Associating Help Messages with Scroll Areas and Grids

After entering a message, you associate the message to a page control. To associate help messages to scroll areas and grids:

1. Open a page definition.
2. Double-click the scroll area or grid to access the properties dialog box.
3. Access the Label tab.
4. In the Embedded Help Popup section, select the Message Catalog Popup option.
5. Enter the Message Set/Number values of the message that you want to appear in the embedded help pop-up page.

The message should appear in the Text field.

6. Verify that the message in the Text field is correct.
7. Click the OK button.

8. Save the page definition.
9. In the browser, access the page and test the embedded help.

---

**Note.** Use the preceding steps to associate Help messages to group boxes, too.

---

**See Also**

*PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Using PeopleSoft Application Pages," Accessing Embedded Help



## Chapter 11

# Creating Page Definitions

Pages are the graphical interface between your users and your application database. As a system designer, you configure or build pages that meet the data requirements of the application and that are easy to use and understand. Using PeopleSoft Application Designer, you can create, modify, and delete page definitions in your PeopleSoft system.

This chapter provides an overview of page design and development and discusses how to:

- Use page development tools.
- Create new page definitions.
- Add page controls.
- Manipulate page controls.
- Set page field properties for controls.
- Modify the rich text editor user interface.
- Create display control and related fields.
- Order and space page controls.
- Design inquiry pages.
- Align page controls.
- Maximize performance.
- Access PeopleCode in page definitions.
- View pages in the browser.
- Produce pages.
- Work with pages in foreign languages.

---

## Understanding Page Design and Development

This section discusses:

- Level-based controls.
- Keys for accessing pages.

- Multiple occurrences of data.
- Prompt fields.
- Derived and work fields.
- Sensitive data.
- Hidden pages.
- Upgrade considerations.
- End-user page personalization.

## Level-Based Controls

You determine your page design based on the type of data that you plan to access and maintain. In some cases, a page references a single record definition; in others, you may want to reference multiple records. To accommodate a variety of page designs, PeopleSoft Application Designer uses *level-based controls*. The three level-based controls are:

- Grids
- Scroll areas
- Scroll bars

You specify the level of these controls by setting the *occurs levels* on the Record tab of the page field properties dialog box for the level-based control that you are configuring. Level-based controls have four levels:

- Level 0

This level directly corresponds to the high-level key information of the underlying record. Level 0 fields are usually physically located at the top of the page, set to be display-only, and display the data that the user entered on the initial search page.

- Levels 1–3

These levels correspond to lower level key information that is subordinate to the Level 0 key values. Level 1 is subordinate to Level 0; Level 2 is subordinate to and nested within Level 1; and Level 3 is subordinate to and nested within Level 2.

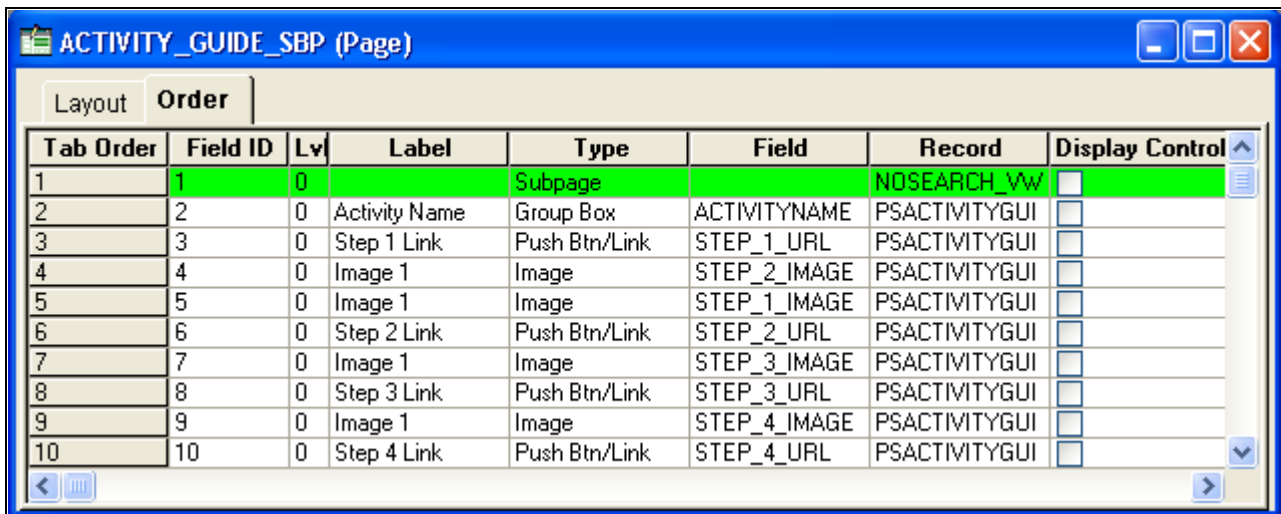
---

**Note.** You can nest level-based controls up to three levels.

---

The first occurs level on a page is Level 0. In general, this level is reserved for the primary key fields that are used to search for data. A page might not contain any level-based controls; in such a case, all fields are set to Level 0. This case is particularly true for secondary pages or subpages that contain few data entry fields, as shown in the following example. The third column, Lvl (Level), that appears in the grid on the Order tab indicates that all fields on the page are at Level 0:





Tab Order	Field ID	Lv	Label	Type	Field	Record	Display Control
1	1	0		Subpage		NOSEARCH_VW	<input type="checkbox"/>
2	2	0	Activity Name	Group Box	ACTIVITYNAME	PSACTIVITYGUI	<input type="checkbox"/>
3	3	0	Step 1 Link	Push Btn/Link	STEP_1_URL	PSACTIVITYGUI	<input type="checkbox"/>
4	4	0	Image 1	Image	STEP_2_IMAGE	PSACTIVITYGUI	<input type="checkbox"/>
5	5	0	Image 1	Image	STEP_1_IMAGE	PSACTIVITYGUI	<input type="checkbox"/>
6	6	0	Step 2 Link	Push Btn/Link	STEP_2_URL	PSACTIVITYGUI	<input type="checkbox"/>
7	7	0	Image 1	Image	STEP_3_IMAGE	PSACTIVITYGUI	<input type="checkbox"/>
8	8	0	Step 3 Link	Push Btn/Link	STEP_3_URL	PSACTIVITYGUI	<input type="checkbox"/>
9	9	0	Image 1	Image	STEP_4_IMAGE	PSACTIVITYGUI	<input type="checkbox"/>
10	10	0	Step 4 Link	Push Btn/Link	STEP_4_URL	PSACTIVITYGUI	<input type="checkbox"/>

Example of page definition Order tab

When you add a level-based control, a scroll area or a grid, to your page, the default occurs level for that control is set to 1 in the properties dialog box. If you place a field in or below that scroll area, it is also set to Level 1 on the Order tab, even if it is another level-based control. If necessary, you can use the Set to Level 0 feature of the horizontal rule control to restart the occurs level on a page.

---

**Note.** You can have any number of level-based controls at the same level.

---

### ***Nested Level-Based Controls***

Nesting controls involve two or more level-based controls on a page, such as two scroll areas, when the second scroll area has an occurs level of 2. You nest controls when the new data that you want to add is a repeating set of data for each entry in your first level-based control. In doing so, you create a hierarchical, or parent and child, relationship between the controls and the processing of the record definitions. The Level 2 control is the child of and is subordinate to the Level 1 control.

For example, on the Compensation page, the Level 2 grid is nested in the Level 1 scroll area. On the Amounts tab of the grid, each compensation action in the Level 1 scroll area has two rate codes:

Compensation

Find

First

1 of 1

Last

Effective Date:

01/01/2000

Go To Row

Effective Sequence:

0

Action:

Hire

HR Status:

Active

Reason:

Payroll Status:

Active

Job Indicator:

Primary Job

Current

Compensation Rate:

1,776.67

AUD

\*Frequency:

M

Monthly

Comparative Information

Pay Rates

Default Pay Components

Contract Change Prorate Option

Pay Components

Customize

Find

First

1-2 of 2

Last

Amounts

Controls

Changes

Conversion

	*Rate Code	Seq	Comp Rate	Currency	Frequency	Points	Percent	Rate Code Group		
1	KABS01	0	1,776.667000	AUD	M				+	-
2	KABSTL	0	1,950.500000	AUD	M				+	-

Calculate Compensation

Example of Compensation page with nested grid in scroll area

This example of the Additional Pay 1 page shows three levels of data, two of which are nesting scroll areas in the Earnings scroll area:

The screenshot displays a web form titled 'Additional Pay 1' with three tabs: 'Additional Pay 1', 'Additional Pay 2', and 'Additional Pay 3'. The form is for employee 'Albion, Charles' with ID 'Z1000' and 'Empl Rcd#': '0'. It features three nested scroll areas:

- Earnings**: Contains a search bar for '\*Code:', a 'Find | View All' link, and navigation buttons 'First', '1 of 1', and 'Last'.
- Eff Date and Default Job Data**: Contains an 'Effective Date' field (07/05/2000), 'Compensation Rate/Frequency' (\$6,000.000000 Monthly), 'Standard Hours' (40.00), and 'Employee Type' (Salaried). It also has a 'Find | View All' link and navigation buttons.
- Or Overrides to Job Data**: Contains fields for '\*Addl Seq #', 'Rate Code', 'Earnings', 'Goal Amt', 'Sep Chk #', 'Earnings End Date', 'Reason' (Not Specif), 'Hours', 'Hourly Rate', and 'Goal Bal'. It also includes checkboxes for 'Disable Direct Deposit', 'Prorate Additional Pay', and 'OK to Pay', and a section for 'Applies to Pay Periods' with radio buttons for First, Second, Third, Fourth, and Fifth.

At the bottom, there are navigation buttons: 'Save', 'Return to Search', 'Next in List', 'Previous in List', 'Update/Display', 'Include History', and 'Correct History'. Below these are links for 'Additional Pay 1', 'Additional Pay 2', and 'Additional Pay 3'.

Example of Additional Pay 1 page with nested scroll areas

In this case, we want to maintain information about the compensation history for a single employee and, for each change or override in compensation status, enter details about that change. The first scroll area, Earnings, associates the earning code with the second, subordinate scroll area, Eff Date (effective date) and Default Job Data, which enables the user to enter the effective date for the status change. The third scroll area, Or Overrides to Job Data, which is subordinate to the second, enables the user to enter the overrides to the default job data that appears in the second scroll area.

For each of the three scroll areas, you can have several rows of data. While you do not see actual scroll bars on your page, the navigation buttons and links in the navigation bars for each scroll area enable you to navigate or scroll through the rows.

### Nested Grids in Scrolls

As shown previously, you can nest a grid in a scroll or scroll area. The following example shows the Other Earnings grid at Level 3, which is nested in the Earnings scroll area at Level 2, which is nested in the Paysheet Details scroll area at Level 1. A nested grid serves the same function as a nested scroll area. It can offer a more compact way of viewing many fields of data, particularly if you use a tabbed grid.

**Note.** You can nest a grid in a scroll area, but you cannot nest a scroll area in a grid or a grid in another grid.

The screenshot displays the 'Paysheet' page with three nested sections:

- Paysheet Details:** Includes fields for Company (BN1), Pay Group (B2M), Pay Period End Date (12/31/1998), and Page (1). It also shows Line 1, Empl ID (B-R5MASET3), Retmaset3,name, and various record and check details.
- Earnings:** Contains checkboxes for OK to Pay, Job Pay, and Override Hourly Rate. It also shows Reg Hours (173.33), OT Hours, Hourly Rate (17.307692), and Reg Salary.
- Other Earnings:** A table with columns for \*Code, Seq Nbr, Rate Code, Hours, Rate, Amount, and Source.

Example of Paysheet page with three levels of nesting

### Levels and Runtime Processing

Levels play an important role in runtime processing. The component processor relies on the level at which you place a field on a page to determine how to process any PeopleCode attached to the field in the record definition.

### Effective Dates and Level-Based Controls

The EFFDT (effective date) field must be the *only* key field that is controlled by level-based controls that you create to help users maintain multiple occurrences of data that is keyed by effective date. Otherwise, the effective date processing for update actions does not function correctly.

### See Also

*PeopleTools 8.52: PeopleCode Developer's Guide*, "Referencing Data in the Component Buffer"

## Keys for Accessing Pages

No matter how much time you spend defining and refining your page, your work is useless unless the user can access it. You must add it to a component, and then you must add that component to a menu. When you add a page to a component, you determine what actions the user must perform to access the page and the keys that are required to retrieve rows of information.

The search record that you define for a component determines the key list; that is, search keys for which users are prompted when they select a page and an action. A search record can be either a view that concatenates information stored in several tables or the underlying table itself. Select the search record that contains all of the key items for the primary record underlying the page.

The keys for which the search dialog box prompts should populate the high-level (Level 0 in the page definition) key controls on a page. These key controls always appear before any level-based control on the page and are typically display-only. A search record might differ from a primary record definition, but it must contain all of the Level 0 keys that you placed on the page.

### ***Using Keys for Views***

When you create a view that should use key values from a page, each key must be:

- Listed as an output column from the view (not just referenced in the WHERE clause of the view).
- Marked as a key field of the view.

For example, suppose that a page has PERSON.PERSONID as a Level 0 field. PERSONID is the key of the PERSON record. A grid on the page uses the PERSON\_ADDRESS\_VW view to display addresses for the current person. The SQL statement for the view cannot only mention PERSON.PERSONID in the WHERE clause; PERSONID must be both an output column of the view and a key field of the view.

### ***See Also***

[Chapter 6, "Creating Record Definitions," page 115](#)

[Chapter 12, "Creating Component Definitions," page 393](#)

## **Multiple Occurrences of Data**

On some pages, you may want a few of the field controls to display multiple rows or occurrences of data. To do this, you can add a level-based control: a scroll area, a grid, or a scroll bar. Users can then add, edit, delete, find, and scroll through multiple occurrences of data in a page control or group of controls using action buttons, links, or the browser scroll bar, depending on how you set the occurs count. Using a scroll area or a grid, rather than a scroll bar, is the preferred page design to show multiple occurrences of data.

For example, this Checklist Table page contains the Assignment Checklist Item grid with an occurs count of 5. The Checklist Item scroll area in which it is nested must have an occurs count of 1 since we can see only one row of data in the scroll area:

**Person Checklist**

Susan Jones Person ID: 0002

**Checklist History** Find | View All First 1 of 1 Last

\*Checklist Date: 08/17/2011

\*Checklist: DEUTER Termination

Comment:

**Person Checklist Items** Customize | Find | First 1-5 of 10 Last

*Sequence	*Item Code	Description	*Status	Link ID		
100	TER01	Cancel Security Access	Initiated			
200	TER02	Update Social Insurance Data	Initiated	<a href="#">Update Social Insurance Data</a>		
600	TER06	Holiday status/ bonus	Initiated			
800	TER08	Process Job Change	Initiated	<a href="#">Job Change Request</a>		
900	TER09	Verify Termination Date	Initiated	<a href="#">Verify Termination Date</a>		

Example of Person Checklist page with a grid of occurs count 5 nested in a scroll area of occurs count 1

To see the next set of five rows in the Assignment Checklist Item grid, a user can click the right arrow button. To see all rows of data at once, in either the grid or scroll area, the user can select the View All link in the navigation bar.

In PeopleSoft Application Designer, you set the occurs count for a level-based control on the General tab in the properties dialog box. You can set the occurs count to any number. You also have the option of selecting Unlimited Occurs Count so that the user sees all rows of data. If the page has more options than can be viewed in the window at runtime, the browser displays a scroll bar enabling the user to scroll to the bottom of the page.

---

**Note.** When an occurs count is set to a number other than 1, the grid displays the specified number of rows. However, when it is set to 1, the grid displays as many rows as possible within the given space allocated for the grid control at design time.

---

## Prompt Fields

You can enable your users to look up the valid values that they can enter in a field. For this, PeopleSoft Application Designer provides prompts, or lookup buttons. Your PeopleSoft applications use the following three types of prompts:

- Drop-down list box

This type of prompt generates a list of values, which appears directly below the prompt field when the user clicks the down arrow that appears beside the field.

The drop-down list box is a predefined control that you place on your page. The Job Indicator field on the Work Location page is an example of drop-down list box.

- Calendar prompt

This type of prompt generates a small calendar, which appears next to the date field, when the user clicks the calendar button. Users can navigate the calendar to find and select a date. The Effective Date, Position Entry Date, and Department Entry Date fields on the Work Location page are examples of calendar prompts. The calendar prompt appears automatically as part of any edit box control that you associate with a date field as long as the Show Prompt Button check box display option on the Record tab is selected.

---

**Note.** The calendar prompt feature is not supported on all browsers.

---

See PeopleTools 8.4x - 8.52 Troubleshooting Browser Limitations available at <http://support.oracle.com>.

- Lookup button

A prompt or lookup button opens a lookup page in the user's browser populated with up to 300 available values for that field. The user can then either select the desired value or refine their search further. For extremely large tables, the system administrator has the option of excluding that table from auto prompting on the Lookup Exclusion Table via PeopleTools, Utilities, Administration.

In PeopleSoft Application Designer, you can only associate a lookup button with an edit box control. The look up button appears automatically as part of any edit box control as long as the Show Prompt Button check box display option on the Record tab is selected and the record field with which you associate the edit box lists values in either the Translate table or a prompt table.

**Work Location** | Job Information | Job Labor | Payroll | Salary Plan | Compensation

**Jaime Salazar**  
Employee

Empl ID: KYLA15  
Empl Record: 0

**Work Location** Find First 1 of 1 Last

\*Effective Date: 01/01/2008  
Effective Sequence: 0  
HR Status: Active  
Payroll Status: Active

Position Number:   
Override

Position Entry Date:   
Position

\*Regulatory Region: MEX  
\*Company: KYF  
\*Business Unit: KY001  
\*Department: 22000  
Department Entry Date: 01/01/2000  
\*Location: KYDF  
Establishment ID:   
Date Created: 12/18/2008

Last Start Date: 01/01/2000  
Expected Job End Date:   
Spain  
Japan

Job Data | Employment Data | Earnings Distribution | Benefits Program Participation

Work Location page showing sample prompt fields

### See Also

Chapter 9, "Using Page Controls," Using Drop-Down List Boxes, page 222

Chapter 4, "Creating Field Definitions," Specifying Date Field Attributes, page 72

Chapter 4, "Creating Field Definitions," Using the Translate Table, page 92

## Derived and Work Fields

You can use a field definition from a derived or work record to store a temporary value that PeopleCode uses to determine the values of other field controls on the page.



For example, for a budgeting transaction in PeopleSoft General Ledger, assume that you have an annual amount that you must spread to multiple accounting periods. You can create a page that includes both a field control from a derived record for the annual amount and an amount control for each accounting period. You then write PeopleCode to derive the amount per accounting period from the annual amount. When a user enters the annual amount in the derived field control, PeopleCode calculates the amount per accounting period and inserts it into each accounting period field control. The annual amount is not stored in the database, but the period amounts are.

## Sensitive Data

You can allocate sensitive data to a single page and limit access to that page to the users who need to update it. Alternatively, you can enter PeopleCode to hide certain fields on pages based on appropriate criteria. The page approach is simpler, however, and can be used in most situations. For secondary pages, this sort of security is more complex to manage.

### See Also

*PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," Hide

## Hidden Pages

Hidden pages are work pages that are associated with derived or work records; they are often used in work groups. You can store all of your work field controls on hidden pages. Create these pages when you want PeopleCode calculations to be performed in the background so that the user does not see them. As a convention, the names of work pages that are delivered with your application names end with the suffix `_WRK`; you should follow the same naming convention. An example name is: `MC_TYPE_WRK`.

Hide a page by selecting the Hidden check box in the component grid as you set up the component definition.

### See Also

Chapter 12, "Creating Component Definitions," Setting Page Attributes, page 399

## Upgrade Considerations

If you adapt a PeopleSoft application, you might affect your use of future PeopleSoft releases. The closer your PeopleSoft system is to the standard product, the easier your upgrades will be. Avoid superfluous or cosmetic changes to data structures in the standard product. Document all of your changes in the comment area on the General tab in the Page Properties dialog box.

## End-User Page Personalization

End users of PeopleSoft applications can personalize various settings on a page, including tab order, the display state of scroll areas and grids, and the state of collapsible sections. Users can also choose that the default page display for a component each time they open that component. In addition, users can choose to share their personalization settings with or copy settings from another user.

The page personalization option is set in the Component Properties dialog box, Internet tab. The Personalize Page Link check box is selected by default to enable the Personalize Page link in the pagebar at runtime for all pages.

### **See Also**

[Chapter 12, "Creating Component Definitions," Setting Internet Properties, page 406](#)

*PeopleTools 8.52: PeopleSoft Applications User's Guide, "Setting User Preferences," Personalizing Pages*

---

## **Using Page Development Tools**

This section discusses how to:

- Use page definition menus.
- Use page definition toolbars.
- Use the page layout grid.

## **Using Page Definition Menus**

As you build pages, use the tools and options in these menus:

- File Menu

In addition to the standard options for opening, deleting, and renaming pages, use this menu to access the definition (page) properties and project properties.

- Edit Menu

Access the page field properties from this menu. To find out what other definitions reference the active page, select Find Definition References. You can also search for a text string in types of PeopleCode or SQL objects.

- View Menu

- Select View Definition to view the underlying definition, such as the subpage or record, for the selected control.

You can view the PeopleCode for the page or the underlying record definition.

- The properties Edit Fill Display, Edit Fill All, and Edit Fill None refer to how controls appear only during design time. They do not affect the page during runtime.
- Use the Show Page Control Inspector tool to view summary data of the selected page field.
- Use Show Grid to display or hide a layout grid on the page that you are designing to help you place controls more accurately.

- **Insert Menu**

Use this menu to insert different types of controls on your page. You can also insert the page on which you are working into a project.

- **Tools Menu**

This menu provides access to various utilities, such as data administration, change control, upgrade, and translate.

- **Layout Menu**

Use this menu to modify the number of pixels between points on the layout grid. Options on this menu also enable you to align page controls, align page control labels, view the processing order of the controls on the page, and test the tab order between fields on your page.

- Use the Align functions to move selected page controls or page control labels to ensure consistency of vertical and horizontal placement among the selected elements. All selected elements align to the last element that you select in the set.
- Use the Auto Size Page function to resize your page to the best height and width that accommodates your page controls.
- Use the View in Browser function to see a very general image of the page in the selected browser without data. Note, however, that PeopleCode is not run on this page rendering, so the page may look significantly different to users if PeopleCode is used to hide or unhide page fields.
- Use the Generate HTML function to create a text file containing the HTML for your page. The file is saved to the TEMP directory of your local drive. Note that the HTML that is generated contains no data or context. It is, therefore, likely to differ from the HTML on a page at runtime.

## Using Page Definition Toolbars

The toolbars in PeopleSoft Application Designer change, depending on the active window. When a page window is active, you see the page definition toolbar and the page control toolbar.

### ***Page Definition Toolbar***



Click the Properties button to open the Page Properties dialog box (Alt+Enter).



Click the Project Workspace button to display or hide the project workspace (View, Project Workspace or Alt+0).



Click the Select Group button to select several controls at once (Edit, Select Group menu item or Ctrl+P).



Click the Default Ordering button to reorder all of the fields on the page based on their relative locations on the page (Layout, Default Ordering).



Click the Test Page button to test the tab order of your page design (Layout, Test Mode or Ctrl+T).



Click the View Page in Browser button frequently throughout the design process (Layout, View in Browser).



Click the Toggle Inspector button to display or hide the floating Definition Inspector window (View, Show Page Control Inspector).



Click the Auto Size button to size a subpage or secondary page to fit around page controls.



Click the Toggle Grid button to display or hide the page grid (View, Show Grid or Ctrl+G).



Click the Left Label Position button to place the page control label for the currently selected field immediately to the left (Layout, Left Label Position or Ctrl+D).



Click an Align (position) button to move the selected page controls so that the edges or center points of the selected page controls share a common axis (Layout, Align, (position)).

See [Chapter 11, "Creating Page Definitions," Aligning Page Controls and Page Control Labels, page 371.](#)



Click an Align Label (position) button to move the selected labels so that the edges or center points of the selected labels share a common axis (Layout, Align, Label (position)).



Click the Auto Size Fields button to resize the selected edit box and drop-down list box page controls so that they are all the same width as the last page control that you select. (Layout, Align, Auto Size Fields)



Click the Show Help button to view the page definition help.

## Page Control Toolbar

The page control toolbar contains a button for each type of page control that you insert. The buttons display images that look like the controls that they insert. You can move this toolbar on your workspace independently of the system toolbar or the other half of the page definition toolbar.





















Frame.



Group box.

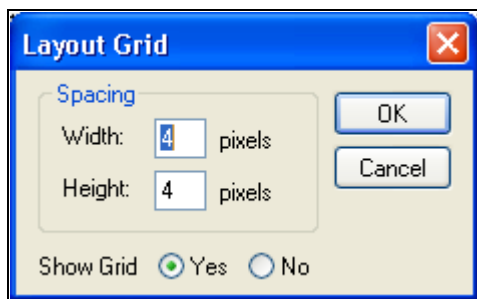


Horizontal rule.

	Static text.
	Static image.
	Grid tab separator.
	Check box.
	Drop-down list box.
	Edit box.
	HTML area.
	Image.
	Long edit box.
	Push button/hyperlink.
	Radio button.
	Tree.
	Grid.
	Scroll area.
	Scroll bar.
	Secondary page.
	Subpage.
	Chart.

## Using the Page Layout Grid

When you open a page definition, evenly spaced dots appear in the workspace. This is the page layout grid that you use to align controls. When you run applications, the grid does not appear on pages. It appears only as a design aid in the page definition. This example shows the Layout Grid dialog box:



Example of Layout Grid dialog box with default values

The default grid spacing is  $4 \times 4$ , which means that each row has one dot per 4 pixels and that the rows are 4 pixels apart on the vertical axis. Most pages in PeopleSoft applications are built using this grid. To change the spacing, open the Layout Grid dialog box by selecting Grid Settings from the Layout menu.

You can turn off the grid view of your page by clicking the Toggle Grid button in the page definition toolbar.

### ***Changing Page Layout Grid Settings***

To change grid spacing:

1. Select Layout , Grid Settings.
2. Enter values for the width and the height, which is the spacing of the dots in pixels.

The smaller the value, the finer the grid. The finer the grid, the more difficult it is to align fields precisely without using the align functionality. Note that if either the width or height is set to one, then the grid will be slightly off.

Select a Show Grid option to activate or deactivate the page grid.

If you copy an existing page and save it by another name, you also copy the grid settings that were used to create the original page. For example, if the original page was built with a  $5 \times 5$  grid, then your new page also will have a  $5 \times 5$  grid, by default. You can change the grid settings anytime while working on a page. Doing so has no effect on the position of a field.

The snap-to-grid functionality is only available when the layout grid is showing.

---

## **Creating New Page Definitions**

To create a new page definition, either clone an existing page or begin with a blank page. Cloning is more efficient because you do not have to recreate controls that are common to all pages. You can also create a template that contains only the controls that you need.

This section discusses how to:

- Clone page definitions.
- Create a blank page.

---

**Note.** When saving new definitions, always make sure to adhere to accepted naming conventions.

---

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33.](#)

## Cloning Page Definitions

To clone a page definition:

1. Select File, Open.

The Open Definition dialog box appears.

2. Specify *Page* for the definition and locate the page that you want to clone.
3. When the system retrieves the page definition, select File, Save As.

If you save after you make modifications, you might accidentally overwrite the old page definition.

4. Enter a new page name.

To create a new page template, name the page so that it appears at the top of your page list.

5. Click OK.

---

**Note.** Record definition names are restricted to a maximum of 15 characters; page names can have up to 18 characters.

---

### See Also

[Chapter 2, "Using PeopleSoft Application Designer," Opening Definitions, page 30](#)

[Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33](#)

## Creating a Blank Page

To create a new page without copying an existing page, select File, New, Page.

When saving the page, make sure that you adhere to the accepted naming conventions.

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33.](#)

---

## Adding Page Controls

You can add page fields to a page in a number of ways. You can copy page fields from one page and paste them onto another, or you can drag-and-drop page fields from one page to another. In addition to these basic methods, this section discusses how to:

- Drag field definitions from a record onto a page.
- Drag record fields from the project workspace onto a page.

- Drag record definitions onto a page.
- Use the page control toolbar.
- Use the Insert menu.

---

**Important!** While PeopleSoft Application Designer does not limit the number of page controls you can add to a page, you might experience unexpected performance issues when you develop pages that contain very high numbers of page controls.

---

## Dragging Field Definitions From a Record Onto a Page

To drag field definitions from a record onto a page:

1. Open an existing record definition by selecting File, Open, Record.
2. Open a new page by selecting File, New, Page.
3. Drag field definitions from the record to the page.

The system selects the page definition control type based on the record field attributes. In the preceding page, the EMPLID (employee identification) field was defined as a prompt table edit in the record, so a Prompt button was added for EmplID when it was dragged onto the page.

4. Double-click the new page field to set the properties.

When you drag field definitions, the properties from the record definition are used to set page field default properties. You can alter the properties using the Properties dialog box.

## Dragging Record Fields From the Project Workspace Onto a Page

To drag record fields from the project workspace onto a page:

1. Open the project that you want to drag from an existing or new page.
2. Drag record fields from the project workspace to the page.
3. Double-click the new page field to set the properties.

---

**Note.** When you drag fields from the project workspace, use fields that are contained in records (as in record.fieldname), not field definitions.

---

## Dragging Record Definitions Onto a Page

You can add an entire record definition to a page at once, which can save a great deal of time when generating pages. You can then arrange the fields as desired on the page.

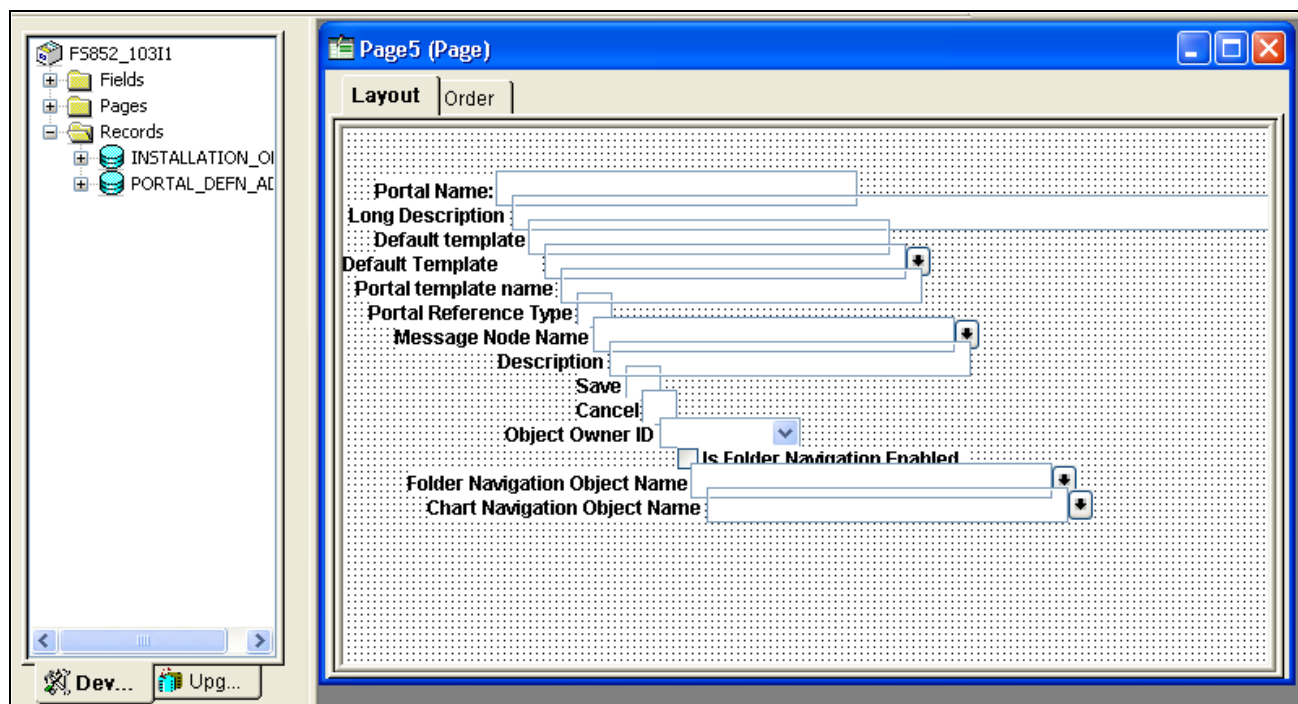
---

**Note.** You can also drag an entire record definition into a grid control that you have placed on a page.

---



See Chapter 10, "Using Scroll Areas, Scroll Bars, and Grids," Inserting and Manipulating Grid Columns, page 283.



Dragging a record definition to a page

To drag record definitions onto a page:

1. Open the project that you want to drag from an existing or new page.
2. Drag the record definition onto the page.

PeopleSoft Application Designer positions the record fields on the page as page fields in a cascading manner. Fields are in the order that they appear in the record with the appropriate control value assigned to each one.

3. Reposition the fields as necessary on your page.
4. Double-click each new page field to set its properties.

## Using the Page Control Toolbar

You can insert page controls by using the page control toolbar. When you click one of the page control buttons, the cursor changes to a cross-shaped icon or hand, depending on which control type you select. You can then move the cursor to where you want to position the control and click once. If you select a control from the toolbar erroneously and want to cancel the selection, press the Esc key and then select the desired page control button.

If the control is a fixed size during insertion, the cursor changes to a cross-shaped icon. If the control type requires that you define the size of the control at the time of insertion, the cursor changes to the monogrammed hand icon. When a hand-shaped icon appears, press and hold the left mouse button as you drag the hand diagonally downward to where you want the lower, right-hand corner of the frame. Release the mouse button.

If the control is not correct the first time that you draw it, you can easily adjust it. The dotted box and black or blue handles surrounding the control indicate that it is selected. Use any of the four directional arrow keys on the keyboard to move the frame one page grid unit in the indicated direction. You can also use Shift+ the arrow key to resize the page control.

## Using the Insert Menu

Select the control that you want to add to your page from the Insert menu. Your cursor changes to a cross-shaped icon or a hand, depending on which control type you select. Move the cursor to where you want to position the control and click once.

---

## Manipulating Page Controls

After you place a control on a page, you have several choices for manipulating it.

This section discusses how to:

- Select controls.
- Resize controls.
- Use the Page Control Inspector.
- Delete controls.
- Move controls on the same page.
- Copy and move controls to another page.
- Position page control labels.

## Selecting Controls

Before you manipulate a control, you must select it. You can do this by clicking the control or by using the Edit menu. A check mark beside the option in the menu indicates which of these modes you are in:

### Select Field

Select one control on the page by clicking on it. To deselect a control, click once outside the dotted box. Deselected is the default mode.

### Select Group

Select a group of controls by drawing a box around the controls that you want to select. You can also activate this option by clicking and dragging over the target fields.

---

**Note.** You are in Select Group mode when the cursor changes to a pointing finger. If the cursor does not change when you select Edit, Select Group, try first selecting an individual control on the page and then selecting Edit, Select Group again.

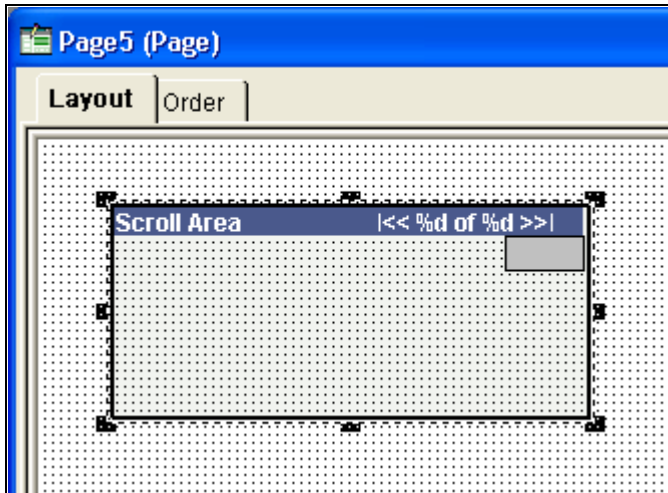
---

**Select All**

Select all of the controls on the active page. The result is the same as using the Select Group option to draw a box around the controls.

**Resizing Controls**

You can adjust the size or shape of any page control that displays selection handles when it is selected:



Resizing a control with selection handles

---

**Note.** Some browsers do not support the resizing of certain control types. In these cases, changing the size in PeopleSoft Application Designer has no effect upon the runtime size of the field.

---

See *Troubleshooting Browser Limitations* available on the My Oracle Support website. [support.oracle.com](http://support.oracle.com).

To resize a control, use one of these methods:

- Select the control and drag a handle to adjust the height or width.

---

**Note.** For some control types, you can adjust only the width.

---

- Hold down the Shift key while pressing the Up Arrow, Down Arrow, Left Arrow, or Right Arrow key.

The frame size is adjusted one grid unit in the indicated direction.

---

**Note.** The position of the top, left corner of the frame does not change; resizing with the keyboard occurs by adjusting the position of the bottom, right corner.

---

**Using the Page Control Inspector**

The Page Control Inspector provides location, size, type, and data source information for the selected page control on the page. The first six values, Left, Width, Right, Top, Height, and Bottom, display the exact bounding coordinates in pixels of the control on your page in PeopleSoft Application Designer. The remaining values help distinguish the selected control from others on the page.

Access the Page Control Inspector from the View menu or by clicking the Toggle Inspector button on the page definition toolbar. When you finish viewing the values for one control, click another control to view its values. You can check multiple controls on a page in quick succession.

The screenshot shows a 'Page Control Inspector' dialog box. The background form has sections: 'Process' with fields for Instance (22222222), Name (NNNNNNNNNNNNNN), Run Status (dropdown), and Type; 'Run' with fields for Run Control ID (AAAAAAAAAAAAAAAA), Location (dropdown), Server (NNNNNN), Recurrence (AAAAAAAAAAAAAAAA), and Current Step (AAAAAAAAAAAAAAAA); and 'Date/Time' with fields for Request Created On (22/22/2222 1) and Run Anytime After (22/22/2222 11:22:22PM PD2), plus a 'Message Log' link.

The 'Page Control Inspector' dialog box contains the following table:

Name	Value
Left	140
Width	136
Right	276
Top	56
Height	15
Bottom	71
Order Number	4
Page Field Type	Edit Box
Label Text	Name
Record	PMN_PRCSLIST
Field	PRCSNAME
Occurs Level	0

Page Control Inspector

**Note.** The HTML that is generated for your page uses table layout, rather than pixel layout. You should preview how your page might look in the PeopleSoft Pure Internet Architecture as you design it. Spacing of fields at runtime may not appear exactly the same as during design time.

While the Page Control Inspector displays values for a grid, no information is available for individual grid columns.

## Deleting Controls

If you remove a control from a page, look for other controls that are related to it, such as display controls or invisible controls used by PeopleCode. You may need to delete them as well or alter their attributes so that they can be standalone fields.

Check for PeopleCode that references this control on the page by opening the field definition and selecting View, Find Definition References. If any references exist, modify or delete them because the control no longer exists on the page.

To delete controls from a page:

1. Select the controls.
2. Select Edit, Clear.

## Moving Controls on the Same Page

To move controls to another position on the same page, use one of these methods:

- Select a control and drag it to the new position.
- Select a control and press the Up Arrow, Down Arrow, Left Arrow, or Right Arrow key.

The control moves one grid unit in the indicated direction.

- Select a group of controls and drag them to the new location.

### ***Moving Controls on Long Pages***

Long pages are too big to be seen in their entirety in the definition workspace. Scroll bars appear on the right and bottom enabling you to move about the page. To move controls that appear in the visible area of the page to an area that is outside the visible area:

1. Select the set of controls by using Edit, Select Group.
2. Click and hold the set of controls.
3. Without releasing the mouse button, scroll the mouse wheel to move up or down the page.
4. Release the button to place the controls.
5. Save the page.

## Copying and Moving Controls to Another Page

To copy a single control to another page, drag the control from one page to the other. For a group of controls, you also can select the controls and use the copy and paste functionality.

To move a group of controls to another page, cut and paste the group onto the new page.

## Positioning Page Control Labels

PeopleSoft Application Designer enables you to change the relative position of most page controls and their labels. This section discusses how to:

- Move page control labels.
- Prevent translation buffer overlap.

### ***Moving Page Control Labels***

To move page control labels, use one of these methods:

- Select multiple labels and either click one of the Align Label (*position*) buttons in the toolbar or select Layout, Align, Align Label(*position*).

PeopleSoft Application Designer aligns the selected labels based on the last label that you select. Alignment is based on the first letter, last letter, or middle letter of the labels; the translation buffer is ignored.

- Select the label and drag it to the new position on the page.
- Select the label and use a directional arrow key on the keyboard to move the label one grid unit in the indicated direction.
- Access the control properties dialog box, select the Label tab, and then select an option in the Position group box (the Location group box for radio button and check box controls). The Custom option is informational only; it does not move the label. All other options move the label to the selected position when you click the OK button.
- Click the Left label position button in the toolbar to move the label to the default left position of the page control.

---

**Note.** Select the control itself when you want to move both the control and its label and retain their relative positioning.

---

### ***Preventing Translation Buffer Overlap***

Often, when you translate your own field labels, the label in the new language requires more space than the English label. When you create or modify pages, you need to accommodate this requirement by leaving additional space between the label and the field control. This space is called the translation buffer. If you do not account for the translation buffer during page design, labels and controls can overlap, which can interfere with readability after the translation.

PeopleSoft Application Designer automatically calculates the translation buffer size as 30 percent of the current label length + the length of 2 upper case 'Ns. The calculation uses the number of characters, but it ignores the individual character itself. The combined character length and translation buffer length is limited to 60 characters for static text labels, 30 characters for RFT\_LONG labels, and 15 for RFT\_SHORT labels.

Based on the calculation, a page control label appears with dashed lines that surround and extend beyond the label, visually indicating the appropriate translation buffer. PeopleSoft Application Designer appropriately moves and adjusts the translation buffer indicators whether you align your labels to the left, right, or center. For most page controls, the translation buffer indicators envelope only the label. However, the translation buffer indicators for push buttons and links enclose the entire control because these labels are embedded in the controls themselves. By default, PeopleSoft Application Designer resizes push buttons and links automatically to accommodate the combined length of the label and the translation buffer.

---

**Note.** Group box, grid, and scroll area controls do not display translation buffer indicators.

---

When you save a page, PeopleSoft Application Designer warns you if the page control label translation buffer overlaps a control. When you dismiss the warning dialog box, PeopleSoft Application Designer brings focus to the problematic page control. If more than one label is inappropriately positioned, the first one receives the focus. When your page contains no translation buffer overlaps, the page saves without warning.

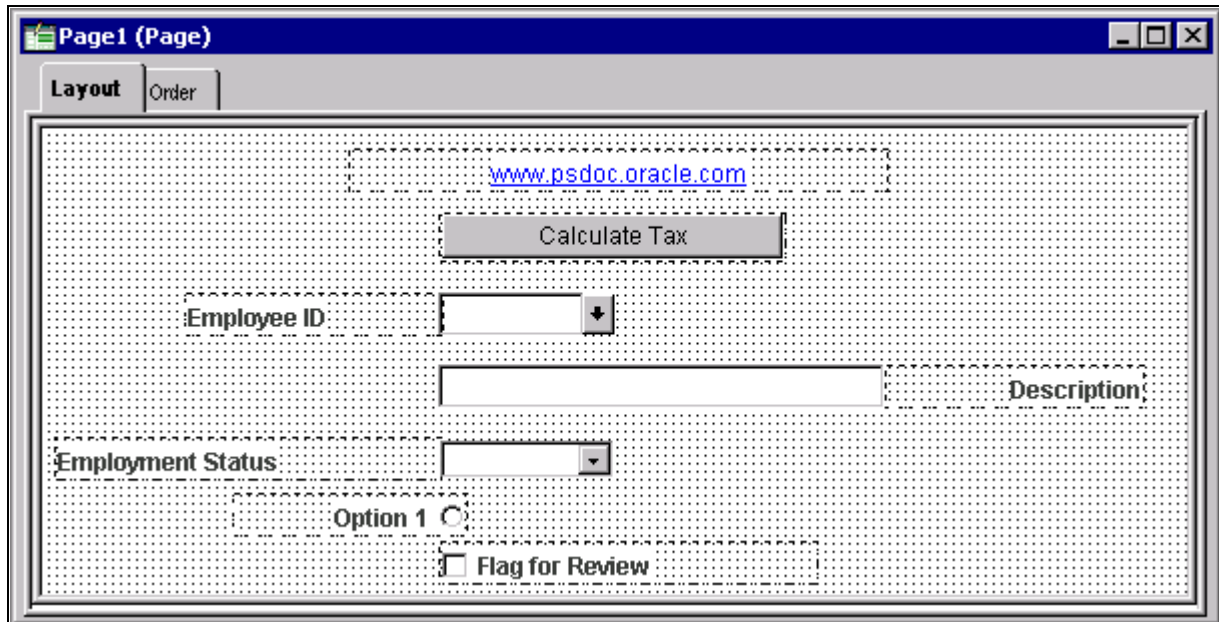
---

**Note.** Translation buffer indicators appear even if you do not translate your labels. Translation buffer overlap, however, is inconsequential to English language pages.

---

### Example: Translation Buffer Indicators

In this example, you see the translation buffer indicators for seven page controls. The link and the push button demonstrate center label alignment. The Employee ID, Employment Status, and Flag for Review labels demonstrate left alignment. The Description and Option 1 labels demonstrate right alignment:



Translation buffers indicators on several page control types showing different label alignment settings

### Example: Translation Buffer Overlap

In this example, the Effective Date label maintains the translation buffer, but the Short Description label does not maintain the translation buffer and overlaps the associated edit box page control:



Field labels demonstrating proper label positioning (Effective Date) and translation buffer overlap (Short Description)

## Setting Page Field Properties for Controls

For each page control that you add to a page you can control how it looks and functions in your application page by setting page field properties. In most cases, you access the property settings by double-clicking the control. Alternatively, you can highlight the control and select Page Field Properties from the Edit menu or the pop-up menu, which is activated by right-clicking the control.

This section discusses how to:

- Set record properties.
- Set label properties.
- Set use properties.
- Set general properties.
- Set option properties.

## Setting Record Properties

For most controls, you must set record properties to assign the control to a specific field in a record definition. No distinct correlation exists between the field types in a record definition and the controls in a page definition.

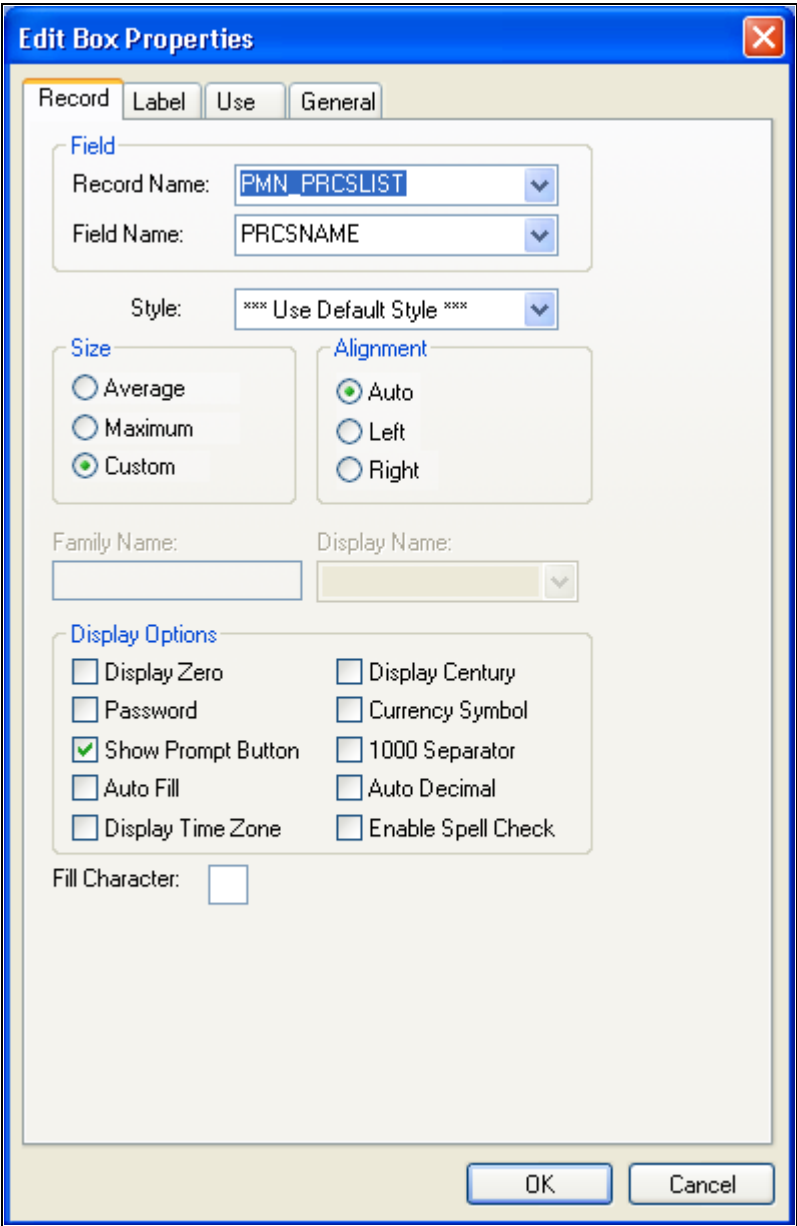
Use this table to select the appropriate page control type to associate with the corresponding field in your record definition:

<i>Record Definition Field Type</i>	<i>Page Definition Control Type</i>
Character	Edit box
Character (Yes or No Table Edit)	Check box Edit box
Character (Translate Table Edit)	Radio button Edit box Drop-down list box
Long Character	Long edit box, HTML area  <b>Note.</b> Do not associate Long Character fields to edit box controls; use a long edit box control. Associating a Long Character field to an edit box control can produce unexpected behavior on the page.
Number	Edit box
Signed Number	Edit box
Date	Edit box
Time	Edit box



<i><b>Record Definition Field Type</b></i>	<i><b>Page Definition Control Type</b></i>
DateTime	Edit box
Subrecord	Not applicable <b>Note.</b> A direct association to subrecords is not available.
Image	Image

To set the record properties for a page control, access the page field properties for the control and select the Record tab:



Edit Box Properties dialog box: Record tab

**Note.** This section applies to both edit boxes and long edit boxes. Other controls may contain only a few, if any, of the properties listed in the following tables. Refer to the discussion of the specific control for more information.

<b>Record Name</b>	Select the record name of the record definition where the field is located. If you already added a field to your page and associated it with a record definition, the system uses the last record definition name that you entered.
<b>Field Name</b>	Select the field name. If the field is a related display field that contains a long name or short name from the Translate Table, use PSXLATDEFN as the record definition name. The field name is either a long name (30 characters) or a short name (10 characters).

**Style**

Set the font and color attributes of your control data. The default style class for an edit box is PSEDITBOX, which controls how the data portion of the edit box appears. Control the color, font, and other characteristics of an edit box by specifying a different style class.

---

**Note.** Certain browsers always display the background of an edit box as white.

---

See "Troubleshooting Browser Limitations" on Customer Connection.

**Size**

After you link a control with a record name and field, the system automatically calculates the page control size. The size is based on the length of the field that is defined in your record definition, the font metrics of the field style, and any formatting options (currency symbol, 1000 separators, and so on) that apply. Because many fonts have variable width characters (for example, a *W* character is wider than a *l* character) three size options are available.

---

**Note.** Some browsers do not support custom sizing for edit boxes, long edit boxes, drop-down list boxes, and push buttons.

---

See *Troubleshooting Browser Limitations* available on the My Oracle Support website. [support.oracle.com](http://support.oracle.com).

**Average**

Select to provide sufficient space to display the field control length in average-width characters.

**Maximum**

Select to provide enough space to display the field control length in maximum-width characters.

**Custom**

Select to define a custom size for edit boxes that are not display-only. If you set a display-only edit box to *Custom*, it appears as its defined length. If your edit box is in a grid, this field wraps.

**Alignment**

Alignment indicates the runtime alignment of data within a page field. These settings have no affect on the relative alignment of page fields.

**Auto**

Select to left-justify the content of character fields and right-justify the content of number and signed number fields. This value is the default.

**Left**

Select to left-justify the content of a field.

---

**Note.** If the field is not display-only, the alignment is always left-justified. If the field is display-only, the system uses the alignment that you specify.

---

**Right**

Select to right-justify the content of the field. Do not use this option in the PeopleSoft Pure Internet Architecture for fields that are *not* display-only.

<b>Family Name</b>	Displays the current family name, which was set when the field was created, if it is applicable to your control.
<b>Display Name</b>	Select an option to override the display format that is associated with a field.
<b>Display Options</b>	
<b>Display Zero</b>	Select if the content of the page control are numeric and you want to display a zero value instead of a blank field. Sometimes, seeing a zero helps users remember that a page control is not blank. For example, in a tax table you might show the lowest tax bracket as starting at zero instead of leaving it blank.
<b>Display Century</b>	Select for date fields to enable users to enter a date with a four-digit century, as in 1999. Dates are always stored with the four-digit century in the database, but only the last two digits appear unless you select Display Century. If this option is not selected, the century is automatically set to the century of your system date.
<b>Password</b>	Select to hide the value that is entered in a page control. This option causes characters in this control to appear online as asterisks (*). The component processor still sees the actual value. Use this option for pages that capture sensitive information, such as personal identification numbers (PINs).
<b>Currency Symbol</b>	Select to display a currency symbol in the field. This selection expands the display length of the field by the length of the currency symbol.
<b>Show Prompt Button</b>	Select to display a prompt button next to the edit box, which enables the user to look up valid values for that field.
<b>1000 Separator</b>	Select if the content of the page control is numeric and you want to insert thousand separators to make the numbers easier to read. The system automatically calculates the number of thousand separators to insert and determines where they should be positioned. This selection expands the display length of the edit box by one character for each separator.
<b>Auto Fill</b>	<p>Select to have the system automatically populate the page control with the character specified in the Fill Character field. The direction in which it populates depends on the alignment that you select. If a field is left-aligned, then it populates from left to right. If a field is right-aligned, it populates from right to left.</p> <p>For example, suppose that you define a six-character control as left-aligned, specify a fill character of 0, and activate Auto Fill. When you type 123 in the control online, the system displays 123000. Similarly, if you changed the alignment to right, the system would display 000123.</p> <hr/> <p><b>Note.</b> Auto Fill can affect the actual value of the control, not just its visual representation.</p> <hr/>

**Auto Decimal**

Select to insert a decimal point automatically if none is provided in the data that is entered into the control. Where the system inserts the decimal point depends on how you define the control.

For example, in a record definition if you define a numeric field with a length of 4.2 (allowing four characters to the left of the decimal and two to the right) and activate Auto Decimal, the system inserts a decimal at the second digit from the right. This option applies only to numeric field controls and affects the actual value of the field control, not just its visual representation.

**Display Time Zone**

Select if you are setting the properties for a Time or a DateTime field and you want to display the related time zone. This setting helps users understand whether the time reflects the base time zone of the database or of some other time zone. This setting does not determine which time zone is actually used, only whether the time zone appears. The record field properties determine which time zone is used.

**Enable Spell Check**

For edit boxes and long edit boxes only, this option enables the user to check spelling on field content. This feature supports 17 languages using 36 dictionaries. Users have the option of selecting the appropriate dictionary to meet the language needs.

The behavior of this feature is as follows:

- **Design time behavior:** The default setting is deselected. When the Enable Spell Check check box is selected, the Show Prompt Button option is deselected and disabled; a field can either be spell checked or validated, but not both. Like the Look-Up button, the spell check button appears to the right of the field and is not visible on the page definition. Be sure to leave extra space to accommodate it when placing other fields on the page.
- **Runtime behavior:** The spell check button appears to the right of the edit box and at the upper right corner of a long edit box. Each click of the spell check button forces a transmission to the application server, opening the Spell Check page. The user then has the option to accept or reject the recommended spelling for the individual word or for all incidences of the term in the field. If no misspellings occur, a message box indicates so without leaving the current page. The Alt+5 hot key invokes spell check while cursor focus is on the spell check field.

The spell check feature uses the current session language when applying a foreign language dictionary, unless the user changes it by means of Personalizations. The system uses PSLANGUAGES to provide the SCLANG default for each supported session language.

**Fill Character**

Enter a character to replace blank spaces in an edit box when the system displays the content of the field. For instance, if the length of a Net Pay field is 8.3 and you specify a fill character of \*, then a value of \$1,250 appears as:

\*\*\*\*\*\$1,250.000

You can use any character as a fill character.

**Example: Auto Decimal**

This table shows how the Auto Decimal option affects a numeric field control with a length of 4.2, depending on what you enter online:

<i>Entered</i>	<i>Displayed</i>
100	1.00
100.	100.00
1	0.01

**Setting Label Properties**

The Label tab enables you to set the label that appears on a control. This label can be the long or short name that is specified in the record definition or in other text. Use this tab to verify the label ID that is used as the internal reference.

Access the page field properties for the control and select the Label tab.

The image shows the 'Edit Box Properties' dialog box with the 'Label' tab selected. The dialog has four tabs: 'Record', 'Label', 'Use', and 'General'. The 'Label' tab contains the following settings:

- Type:** Radio buttons for 'None', 'Text', 'RFT Short' (selected), and 'RFT Long'.
- Label Text:**
  - Text:** A text box containing 'Name'.
  - Label ID:** A drop-down list showing 'PRCSNAME "Name"'.
  - Context:** An empty text box.
  - Insert Line Feed:** A button.
  - Style:** A drop-down list showing '\*\*\* Use Default Style \*\*\*'.
- Alignment:** Radio buttons for 'Left', 'Centered', and 'Right' (selected).
- Position:** Radio buttons for 'Left' (selected), 'Top', and 'Custom'.
- Display Options:**
  - ☐ First occurs only
  - ☒ No colon

At the bottom right are 'OK' and 'Cancel' buttons.

Edit Box Properties dialog box: Label tab

---

**Note.** This section applies to the following controls: edit boxes, long edit boxes, check boxes, drop-down list boxes, group boxes, radio buttons, and static text.

---

### Type

Select the type of label. If you want the label to remain blank, you must first set up a blank label in the field definition that you are using. The blank label then appears as an option in the Label ID drop-down list box.

#### None

Select to display no control label on the page. Use this option for controls such as related displays.

<b>Text</b>	Select to display text that you enter in the Text field. The default text is the long name for the field from the associated record definition.  For translation purposes, avoid using this option if possible.
<b>RFT Short</b> (record field table short)	Select to display the RFT short name for the field from the associated record definition.
<b>RFT Long</b> (record field table long)	Select to display the RFT long name for the field from the associated record definition. This value is the default.

### ***Label Text***

<b>Text</b>	Enter the text exactly as it should appear on the page. The default is <i>Dummy Name</i> until you assign a record and field to the control. This field is only enabled if you selected Text as the type. Otherwise, the system displays the label text assigned to the field for the Label ID you select.
<b>Label ID</b>	Select from the drop-down list box the label you want to appear for the page field. The available choices are based on the Label ID column and the Long Name column for the selected field definition.  The default setting for this option is <i>Use Default Label</i> . This default is determined by the label you select as the default for the field definition.
<b>Context</b>	Displays text to assist the translation team determine the context of the field label to be translated. The context is determined by the field name and label ID combination. If no contextual information exists in the database, then no text appears in this field.  This property appears on edit box, long edit box, drop-down list box, push button, image, chart, tree, group box, and check box page controls.  Application users define the contextual information on the Context Information page.  See
<b>Insert Line Feed</b>	Click to split your control label into multiple lines at the position of the cursor in the Text field. A thick vertical bar character appears in the Text field. When you close the properties dialog box, the label is split into multiple lines. This option is only active when you select <i>Text</i> for the type.
<b>Style</b>	Select an option to control the color, font, and other characteristics of a label. If you do not select a label style that is different than the default, <i>Use Default Style</i> , then the system uses the label style specified on the style sheet on the PeopleTools Options page.



## Alignment

The label alignment in the page field properties controls the alignment for both page fields placed on a page and page fields as grid columns. For a grid column, the label becomes the column heading and is aligned within the column boundaries as specified by this attribute, either left, centered, or right.

<b>Left</b>	Select to align the label to the left-of-center horizontally. This value is the default.
<b>Centered</b>	Select to center the label horizontally.
<b>Right</b>	Select to align the label to the right-of-center horizontally.

---

**Note.** Push buttons with images or text are automatically center aligned when in a grid. Push buttons with images or text are automatically left aligned when *not* in a grid.

---

## Position

These position options are available for edit boxes, long edit boxes, and drop-down list boxes only.

<b>Left</b>	Positions label immediately adjacent to the left side of the field. Left is the default position of the label for fields newly added to the page. If you move the label on the page from the left position, PeopleSoft Application Designer changes the selection to Custom.
<b>Top</b>	Positions label above the field. Selecting this option automatically moves the label to the top of the field. If you later move the label on the page from the top position, PeopleSoft Application Designer changes the selection to Custom.
<b>Custom</b>	Indicates that the label was manually moved on the page. Labels can be adjusted using the keyboard arrows. Selecting this option does not automatically move the label on the page.  See <a href="#">Chapter 11, "Creating Page Definitions," Positioning Page Control Labels, page 335.</a>

To set labels to the left position for individual page fields:

1. Select the control that you want to affect.
2. Select Layout, Left Label Position.

The Left Label Position option is only enabled if the label is set to the custom position. The system moves the label for the selected control to the left position. The position indicator in the page field properties does not change from *Custom*. Note that you may need to readjust the positioning of other fields and labels to accommodate this change.

---

**Note.** If you move a page field label manually on the page, the label position in the page field properties dialog box automatically updates to *Custom*.

---

## Display Options

<b>First occurs only</b>	Select to display the label only with the first occurrence of a child page field in a scroll area.
<b>No colon</b>	Select this option if you do not want a colon to appear at the end of the label text. This option is selected by default for edit boxes, long edit boxes, and check boxes.

## Special Label Considerations

If you add a related display page control with a label type of *None*, enter an information-only label that identifies the related field but that is not identical to the label on the control field. A little documentation now can save you time in the future if you modify this page again. For example:

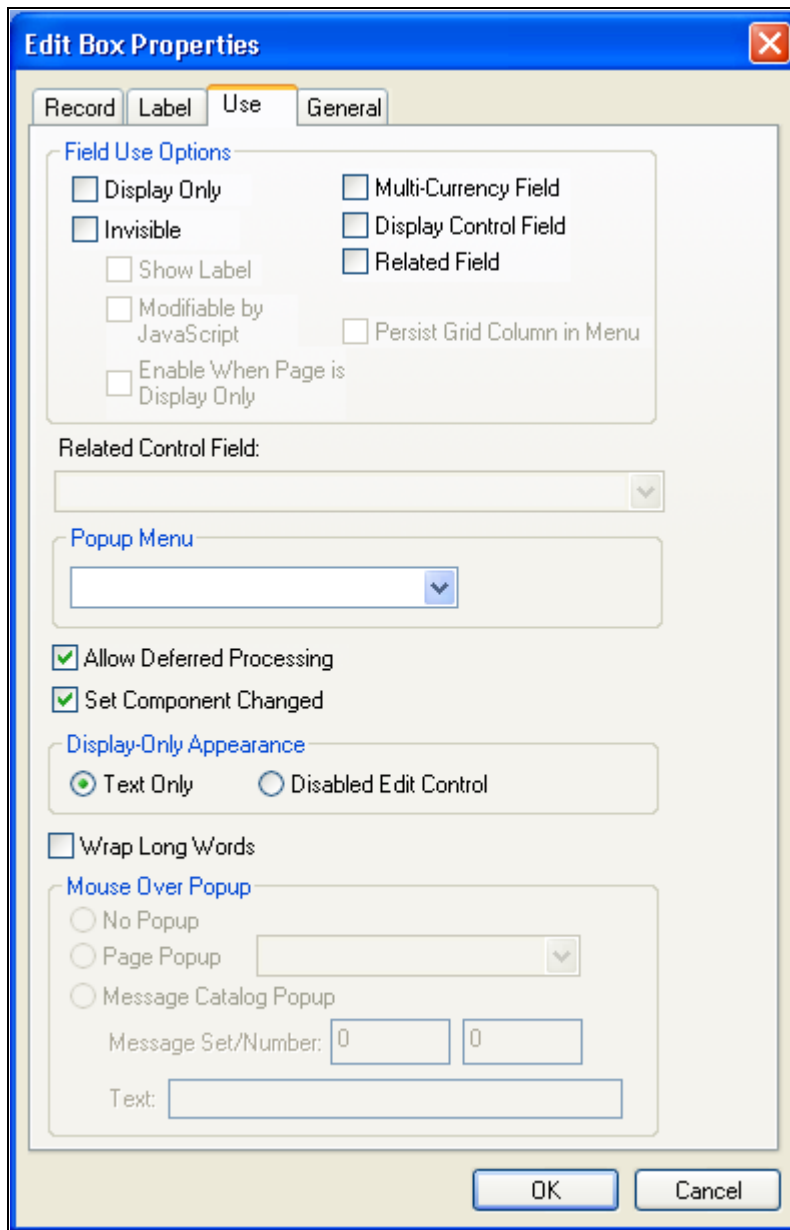
<b>Control Label</b>	<b>Related Field Label (Information-Only)</b>
Department	Department Description
Jobcode	Jobcode Description
Regular/Temporary	Regular/Temporary XLAT (XLAT is from the Translate Table)

When defining labels for columnar controls in a multi-occurrence scroll, you can apply several techniques. The first technique is to select First occurs only. This selection limits the appearance of the label to the first scroll occurrence on the control, allowing the label to be used as a column heading. The second technique is to select *No colon* to omit the colon from the label (column headings typically do not need a colon).

## Setting Use Properties

The Use tab defines how a field functions on a page. Use this tab to disable or hide fields; to define the display and related control fields and the processing of the control; and to define a field as a mouse over pop-up field.

Access the page field properties for the page control and select the Use tab.



Edit Box Properties dialog box: Use tab

---

**Note.** These controls contain the same or similar use properties: edit boxes, drop-down list boxes, check boxes, images, long edit boxes, push button/hyperlinks, and radio buttons.

---

### ***Field Use and Other Options***

Note that long edit boxes have slightly different field use options.

#### **Display Only**

Select to prevent users from modifying the content of the control during application data entry. The system automatically selects Display Only when you select Related Field.

<b>Invisible</b>	<p>Select to make the control physically present on a page but invisible to users. Typically, you add an invisible control because it is required for a PeopleCode program that is associated with the page. An invisible control can also be used as a display control field.</p> <p>When you select the Invisible check box, the Show Label, Modifiable by JavaScript, and Enable When Page is Display Only fields are enabled.</p>
<b>Show Label</b>	<p>Select to make the control label visible while the control itself is invisible. This option is useful if you add an invisible display control field to a page to show its related description. Show Label is enabled only if Invisible is selected.</p>
<b>Modifiable by JavaScript</b>	<p>This feature is security related and should <i>always</i> be deselected unless you are familiar with modifying an invisible field using JavaScript in an HTML area. If the Invisible check box is deselected, then Modifiable by JavaScript is deselected and unavailable for entry. If the Invisible check box is selected, then the Modifiable by JavaScript check box is deselected by default.</p>
<b>Enable When Page is Display Only</b>	<p>Select to enable hyperlink actions on read-only pages. This check box is active when the Invisible check box is selected. This check box is deselected by default.</p> <hr/> <p><b>Note.</b> This property appears on Edit box, Long Edit box, and Image controls.</p> <hr/>
<b>Expand Field when Page is Display Only</b>	<p>Select to automatically resize a disabled (display-only) long edit box so that all text within the field is visible at runtime. The field automatically enlarges to accommodate text. This check box is deselected by default. When selected, this option enables the Wrap Long Words option.</p>
<b>Multi-Currency Field</b>	<p>Select to identify the control as associated with multicurrency processing. This selection causes the field to appear at runtime only if Multi-Currency is selected on the PeopleTools Options page.</p>
<b>Display Control Field</b>	<p>Select to indicate that the field controls another field at the same page level. The controlled field is a related field. For example, on many pages Department ID is the display control field and Department Name is the related field. The system uses Department ID to retrieve Department Name from the Department table to display on the page.</p>

**Related Field**

Select to specify that the content of this control is controlled by another field on the same page level, one that you selected as a display control field. When you select this check box, the system automatically selects the Display Only option. It also makes the Related Control Field drop-down list box available for you to select the related control field.

When you select Related Field, Display Only is automatically selected. If the related field is located on a record definition with multiple keys, relate the display field to the lowest order key field that both record definitions have in common, with the exception of EFFDT and EFFSEQ, in the control record definition. The system searches for the higher order keys by matching field control names on the current level and all higher levels for the other keys.

If the related display field contains a long name or short name from the Translate Table, use PSXLATDEFN as the record definition name.

**Enable When Page is Display Only**

Select this check box to enable the user to select radio buttons and check boxes on runtime pages to perform certain tasks and processes despite the entire page being in display-only mode.

---

**Note.** This property appears on Check Box and Radio Button controls only.

---

**Wrap Long Words**

Select to wrap long words that do not contain spaces, such as long URLs, in a page field at runtime for Microsoft Internet Explorer only. The word wraps to the length of the field set at design time. Other browsers automatically stretch the width of the page to accommodate the longer text. This option has no effect at runtime to applications running on browsers other than Microsoft Internet Explorer. This setting is deselected by default.

The Wrap Long Words option is available for edit boxes, long edit boxes, and drop-down list boxes only. You can apply it to standalone page fields and grid columns. For edit boxes and drop-down list boxes, this option is disabled if Disabled Edit Control is selected. For long edit boxes, this option is available only if the Expand Field When Display Only option under Field Use Options is selected.

---

**Note.** The system does not restrict an edit box or drop-down list box that is configured as Display Only and Text Only to one line of height. The text is wrapped to the next line as necessary.

Also, the Wrap Long Words property does not affect the word wrapping behavior of shorter words that can be accommodated by the page field size at runtime. This text automatically wraps at word boundaries in accordance with the page field size, regardless of whether this option is set.

---

**Persist Grid Column in Menu** Select to include the field as a list box item in the persistent search results grid. You must also select the Persist in Menu grid property for the grid that contains the field.

---

**Note.** This property applies only to fields that appear in grids on pages that are used as custom, application-specific search pages.

---

This property is disabled on check boxes and radio buttons.

You must enable recent search results caching at the system level in the web profile for this property to take effect.

### **Related Control Field**

When you select Related Field, a list of all controls on the page that are marked as display control fields appears in this drop-down list box. Select the field to which this control is related. You must define the use of the initial control field before it appears as an option in the Related Control Field drop-down list box. For example, after the Pay Group field is designated as a display control field, it appears in the Related Control Field drop-down list box for the control that you set as the related field.

Note that the number that appears before each entry in the Related Control Field drop-down list box is the page field order number associated with that entry.

### **Display-Only Appearance Option**

The Display-Only Appearance option attributes, Text Only option, and Disabled Edit Control option apply only to edit boxes and drop-down list boxes. These properties, in conjunction with other attributes you apply to the field, such as Display Only on the Use tab, or the Gray function or the Enabled property in PeopleCode to disable the field, and whether or not the field is in a grid control the base style that the system uses to render these controls at runtime.

The default base style for a normal edit box, meaning editable and enabled (as opposed to display-only or disabled), is PSEDITBOX. The default base style for drop-down list boxes is PSDROPDOWNLIST. The base style then changes depending on the attributes that you select and where you place your field, as shown by the default base style examples in the following table.

If you choose a custom base style for the field on the Record tab of the control properties, then the system uses the associated display-only and disabled variants of the style as appropriate to create the additional looks. For example, if you select XYZ base style and set the field as display-only, then the related XYZ\_DISPONLY style is used if it exists; otherwise, the base XYZ style is used.

<b><i>Display-Only Appearance Option</i></b>	<b><i>Normal Edit Field</i></b>	<b><i>Display-Only Edit Field</i></b>	<b><i>Disabled Edit Field</i></b>
Text Only	<i>Non-Grid:</i> PSEDITBOX <i>Grid:</i> PSEDITBOX	<i>Non-Grid:</i> PSEDITBOX_DISPONLY <i>Grid:</i> PSEDITBOX_DISPONLY	<i>Non-Grid:</i> PSDISABLED <i>Grid:</i> PSEDITBOX_DISPONLY

<b>Display-Only Appearance Option</b>	<b>Normal Edit Field</b>	<b>Display-Only Edit Field</b>	<b>Disabled Edit Field</b>
Disabled Edit Control	<i>Non-Grid:</i> PSEDITBOX <i>Grid:</i> PSEDITBOX	<i>Non-Grid:</i> PSDISABLED <i>Grid:</i> PSDISABLED	<i>Non-Grid:</i> PSDISABLED <i>Grid:</i> PSDISABLED

Note that for Text Only disabled edit boxes using the default PSEDITBOX base style, the system applies PSDISABLED to non-grid fields and PSEDITBOX\_DISPONLY to fields in a grid. As a result, these fields render differently at runtime, as shown in the following example. If the field is outside of a grid, for example, selecting the Display Only field use option with the Text Only appearance option displays the field value as disabled text within a grayed border, whereas if the field is located in a grid with the same attributes, the field value renders as display only text without a border.

The screenshot displays two runtime examples of the Display-Only Appearance settings. The top example, labeled 'Appearance = Text Only', shows three input fields: 'Normal Edit' (containing 'Normal'), 'Display Only Edit' (containing 'display only'), and 'Disabled Edit' (containing 'disabled'). The bottom example, labeled 'Appearance = Disabled Edit', shows the same three input fields. Both examples are part of a larger interface with a 'Find' button, 'View All' link, and a grid icon.

Example of Display-Only Appearance settings at runtime using the base PSEDITBOX style

See [Appendix C, "Using Default Style Classes," Default Style Classes, page 591](#).

**Note.** Depending on the style sheet that your application uses, the appearance of your pages and fields might deviate slightly from the appearance of the page and fields in the previous example.

### Mouse Over Popup Options

Select the Display Only check box to enable the Mouse Over Popup options. Select from these mouse over pop-up options:

#### No Popup

Select to have no pop-up page. This value is the default.

#### Page Popup

Select to associate the edit box with a pop-up page that you choose from the drop-down list box. Only pop-up pages appear in the Page Popup drop-down list box.

**Message Catalog Popup**

Select to associate this edit box with a pop-up page that displays a message from the message catalog. You can design and implement message catalog pop-up pages to be context sensitive.

See *PeopleBooks and the Online PeopleSoft Library*, "Managing the PeopleSoft Online Library and PeopleBooks," Integrating Context Sensitivity.

See Chapter 9, "Using Page Controls," Enabling Message Catalog Pop-up Pages, page 231.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," Message Catalog.

**Message Set/Number**

Enter the message set number and the message number of the message whose text you want to appear in the pop-up window.

**Text**

This field displays the message of the message catalog entry that you enter in the message set/number fields.

---

**Note.** If you associate a mouse-over pop-up page to an edit box, then do not attempt to make the edit box editable at runtime, for example by using PeopleCode; the operation will fail.

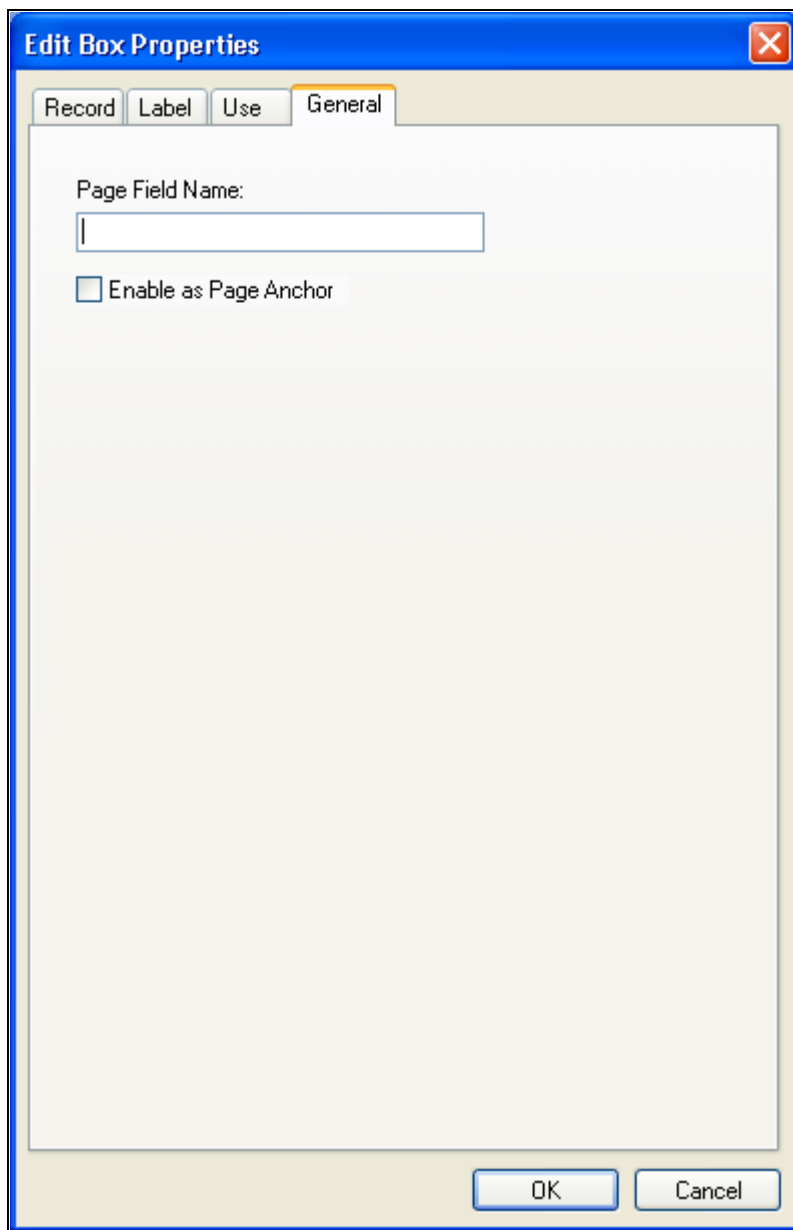
---

## Setting General Properties

The General tab enables you to specify an optional internal page field name that is referenced by the page.

Access the page field properties for the control and select the General tab.





Edit Box Properties dialog box: General tab

---

**Note.** For most controls, the General tab contains only the Page Field Name and Enable as Page Anchor settings. Grids and scroll areas have additional settings on the General tab that are described later. The Enable as Page Anchor check box is available for all controls that have a page field name.

---

### Page Field Name

Enter a name for the page field that you are creating. This is a system-only setting. It does not appear as the label for the page field. The page field name enables you to refer to a particular field on a page using PeopleCode. This setting is optional unless you select the Enable as Page Anchor check box.

**Enable as Page Anchor**

Select to apply an anchor tag to the current page field on the page. If you select this check box, you must add a page field name to identify this field when setting the related control for the link. This check box is deselected by default.

See [Chapter 9, "Using Page Controls," Specifying Type Properties for Push Buttons or Links, page 252.](#)

## Setting Options Properties

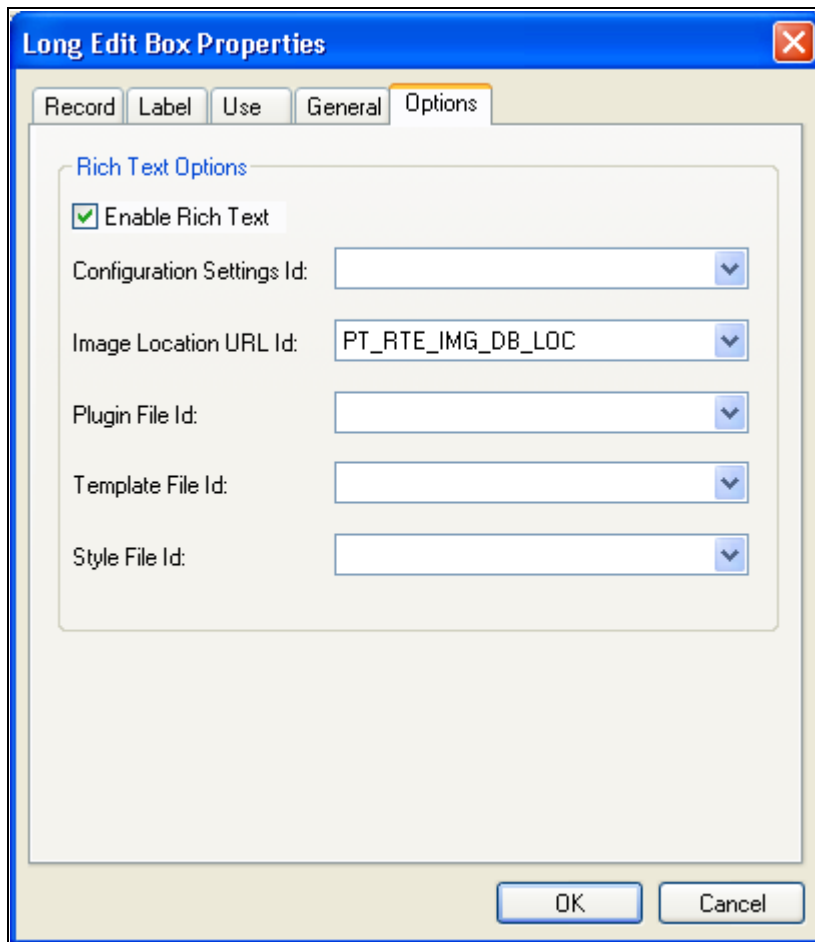
The Options tab enables you to add rich text editing capabilities to the long edit box.

Access the page field properties for the long edit box page control and select the Options tab.

---

**Note.** This section applies to long edit boxes only.

---



Long Edit Box Properties dialog box: Options tab

---

**Note.** At design time, standard long edit boxes and rich text-enabled long edit boxes look the same.

---

**Enable Rich Text**

Select to enable rich text editor functionality for the long edit box. Selecting this check box enables the remaining fields on this tab.

---

**Note.** When you select this option the Multi-Currency and Wrap Long Words properties on the Use tab are disabled; and the Allow Deferred Processing property on the Use tab is selected and disabled.

---

**Configuration Settings Id**

Select a value to use a custom configuration file that modifies the editor toolbar. By default, this field is blank and the system uses the default configuration file with the default toolbar settings.

HTML definitions that are prefixed with *PT\_RTE\_CFG* appear in this drop-down list box.

See [Chapter 11, "Creating Page Definitions," Creating Custom Configurations, page 359](#).

**Image Location URL Id**

Select a value to indicate the storage location of any images inserted as part of the text in the long edit box. The default value for this field is the *PT\_RTE\_IMG\_DB\_LOC* URL.

URL IDs that are prefixed with *PT\_RTE\_IMG* appear in this drop-down list box. The URLs must be stored in the application by using the URL Maintenance page.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," URL Maintenance.

---

**Note.** The URL format for file attachments stored in the database is record://recordname, as in record://PTRTDB.

---

**Plugin File Id**

Select a value to enable plug ins for this instance of the rich text editor. By default, this field is blank.

HTML definitions that are prefixed with *PT\_RTE\_PLUGIN* appear in this drop-down list box.

**Template File Id**

Select a value to enable an alternative template definition for this instance of the rich text editor. By default, this field is blank.

HTML definitions that are prefixed with *PT\_RTE\_TMPL* appear in this drop-down list box.

**Style File Id**

Select a value to enable an alternative style sheet definition for this instance of the rich text editor. By default, this field is blank.

HTML definitions that are prefixed with *PT\_RTE\_STYLE* appear in this drop-down list box.

### **Data Conversion Considerations**

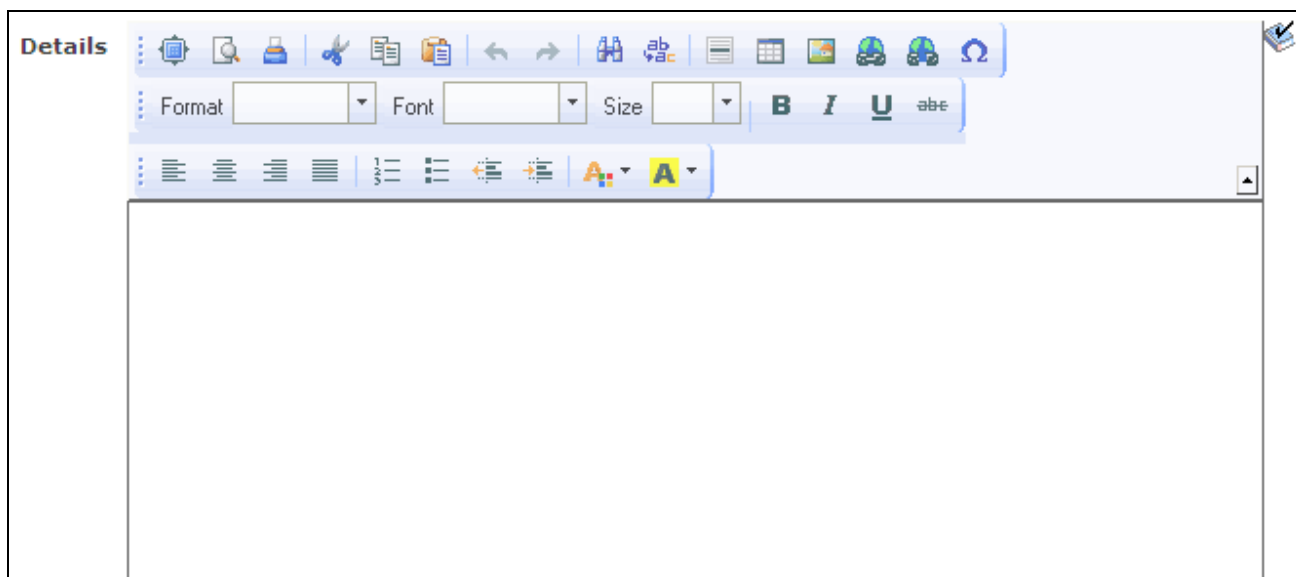
The rich text editor is an HTML editor that converts rich text to HTML, which the application stores in the database. When you access your data, the editor decodes the HTML at the time that it appears on the transaction page. Existing data is stored as plain text, which can cause unexpected warnings the first time the data is accessed after rich text support is enabled on a long edit box page field. Your test plan should include either updating the data to prevent the warnings or communicating with users to expect this warning.

To update the data, you must access the page; change the data in some way, such as adding a Space; and then save the component. The editor automatically encodes and updates the data.

---

## **Modifying the Rich Text Editor User Interface**

The rich text editor extends the capability of a long edit box so that, at runtime, it can display rich text as you see here:



Runtime long edit box showing rich text functionality

The rich text editor enables you to add rich formatting of text content, including common structural treatments such as lists; formatting treatments such as bold, underline, and italic text; drag-and-drop inclusion; paste-as-text; and sizing of images. The rich text editor toolbar is extensible through the use of HTML definitions that define your configuration, which enables you to design and implement an interface that is specific to the needs of your users.

This section discusses how to:

- Create custom configurations.
- Create custom templates.
- Create custom styles.
- Create custom plug-ins.

- Modify the default image storage location.

---

**Important!** Use caution when modifying the rich text editor interface; it is recommended that you never include or enable the View Source button in the toolbar.

---

### See Also

*PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Using PeopleSoft Application Pages," Working With Rich Text Editor Fields

<http://docs.cksource.com/>

## Creating Custom Configurations

The default configuration filename is *config.js*. You can locate this file in the *PS\_HOME\webserv\domainname\applications\peoplesoft\PORTAL.war\sitename\ckeditor* directory.

To create a custom configuration:

1. Copy the code in the *config.js* file.
2. Open Application Designer and create a new HTML definition.
3. Paste the code into the HTML definition.
4. Save the HTML definition. Prefix the definition name with *PT\_RTE\_CFG* for example, *PT\_RTE\_CFG\_MYCONFIG*.

---

**Note.** Only HTML definitions with the *PT\_RTE\_CFG* prefix appear in the Configuration Settings Id drop-down list box.

---

5. Modify the code in the HTML definition as necessary and save. The code might look like this example:

```
CKEDITOR.editorConfig = function( config )
{
    config.toolbar =
    [
        [ 'Templates', 'Styles', 'Cut', 'Copy', 'Paste' =>
        , 'Bold', 'Italic', 'Underline', 'Strike' ],
    ];
};
```

In this example, the Cut, Copy, Paste, Bold, Italics, Underline, and Strike buttons and the newly created custom templates and the styles appear in the toolbar at runtime.

6. To use the new configuration, open the Long Edit Box Page Field Properties dialog box; select the Options tab; click the Configuration Settings Id drop-down list box; and then select the HTML definition name from the list of values that appear.

See [Chapter 11, "Creating Page Definitions," Setting Page Field Properties for Controls, page 337.](#)

## Creating Custom Template Files

The rich text editor delivers templates that help you format the text in predetermined ways. The default template file is *default.js*. You can locate this file in the *PS\_HOME\webserv\domainname\applications\peoplesoft\PORTAL.war\sitename\ckeditor\plugins\templates\templates* folder.

To create a custom template file:

1. In Application Designer, create an HTML definition and save it as *PT\_RTE\_TMPL\_TEMPLATE*. The definition name should always be prefixed with *PT\_RTE\_TMPL,\_* and the HTML must include 'pt\_rte\_template' as you see in this example:

```
CKEDITOR.addTemplates('pt_rte_template',{imagesPath:CKEDITOR.getUrl⇒
(CKEDITOR.plugins.getPath('templates')+'templates/images/'),templates:[{title⇒
'Image and Title',image:'templatel.gif',description:'One main image with a title⇒
and text, which surrounds the image.',html:
<h3><img style="margin-right: 10px" height="100" alt="" width="100" align="left"⇒
/>Type the title here</h3>
<p>Type the text here</p>' } ]});
```

---

**Note.** The name 'pt\_rte\_template' is the fixed name, which has to be set for any custom template that you create. The template name should be exactly the same as mentioned above.

---

Save the HTML definition.

---

**Note.** It is recommended that if you create a custom template file, that you create a custom configuration file for it to reference.

---

2. To use the new configuration, open the Long Edit Box Page Field Properties dialog box; select the Options tab; click the Template File Iddrop-down list box; and then select the HTML definition name from the list of values that appear.

See [Chapter 11, "Creating Page Definitions," Setting Page Field Properties for Controls, page 337.](#)

3. Click the OK button.
4. Save the page definition.

---

**Note.** When associated to a long edit box, all templates in an HTML definition are available to all users. To limit the available templates for specific rich text enabled long edit boxes, create a new template HTML definition to contain only the templates that you would like for the application user to see and associate that definition to the field.

---

## Creating Custom Style Files

The rich text editor delivers styles to assist you in entering data using specific styles that combine color, font and format. The default configuration file is *default.js* and is in the *<PS\_HOME>/webserv/.../styles* folder.

To create a custom style file

1. In Application Designer, create an HTML definition and save it as PT\_RTE\_STYLE\_MYSTYLE.

The HTML Object name used for defining styles should always be prefixed by *PT\_RTE\_STYLE*.

This is sample code for an HTML style definition:

```
CKEDITOR.addStylesSet('pt_rte_style',[{name:'Blue Title',element:'h3',styles:⇒
{color:'Blue'}}]);
```

---

**Note.** In this style example, the name 'pt\_rte\_style' is the fixed name, which has to be set for any custom style that you create. The style name should be exactly the same as mentioned above.

---

2. To use the new configuration, open the Long Edit Box Page Field Properties dialog box; select the Options tab; click the Style File Iddrop-down list box; and then select the HTML definition name from the list of values that appear.

See [Chapter 11, "Creating Page Definitions," Setting Page Field Properties for Controls, page 337.](#)

3. Click the OK button.
4. Save the page definition.

---

**Note.** When associated to a long edit box, all styles in an HTML definition are available to all users. To limit the available styles for specific rich text enabled long edit boxes, create a new style HTML definition to contain only the styles that you would like for the application user to see and associate that definition to the field.

---

## Creating Custom Plug-In Files

A plug-in enables you to create an application-specific toolbar button.

See [Appendix I, "Creating Custom Plug-In Files for the Rich Text Editor," page 639.](#)

## Modifying the Default Image Storage Location

By default, when you upload images the system stores them in the PTRTDB table. The default URL ID is PT\_RTE\_IMG\_DB\_LOC. To create a different storage location and URL ID, follow these steps:

1. Create a new record definition.
2. Insert the *FILE\_ATTDET\_SBR* subrecord.
3. Save and then build the record definition.
4. Clone these record definitions:
  - PTRTDB\_BMP
  - PTRTDB\_GIF
  - PTRTDB\_JPG

5. Save the record definitions, name them like this: <myrecord>\_BMP, <myrecord>\_GIF, and <myrecord>\_JPG.
6. For each record, open the SQL Editor and change the FROM statement in the SQL to this:  

```
FROM PS_<your record name from step 3>
```
7. Save and build each view.

---

**Note.** FTP storage is not supported.

---

---

## Creating Display Control and Related Fields

This section discusses how to:

- Use display control and related fields.
- Create related edit fields.
- Create invisible control fields.

## Using Display Control and Related Fields

A related display field is for display purposes only—it always references a row that's not being updated from the current page. The display control and related display fields must both be at the same level. In the Work Location page, the Regulatory Region, Company, Business Unit, Department, and Location fields all show related display fields to their right. Search level fields, such as those shown above the Work Location group box, are display-only fields.



Work Location		Job Information		Job Labor		Payroll		Salary Plan		Compensation	
<b>Caroline SABBARICH</b>				Empl ID:		GF100ME280					
Employee				Empl Record:		0					
<div> <div>Work Location</div> <div>Find First 1 of 1 Last</div> <div>Go To Row + -</div> </div>											
*Effective Date:		01/01/2000		Effective Sequence:		0		*Action:		Hire	
HR Status:		Active		Reason:				*Job Indicator:		Primary Job	
Payroll Status:		Active								Current	
Position Number:				Override Position Data							
Position Entry Date:				Position Management Record							
*Regulatory Region:		FRA		France							
Company:		GF2		Societe Francaise production							
*Business Unit:		GF001		French Business Unit QA							
*Department:		GFDEV02		Department Develop QA02PP							
Department Entry Date:		01/01/2000									
*Location:		GFPARIS04		PARIS 04							

### Work Location page

Because of the nature of control fields and related fields, PeopleSoft recommends that you:

- Place them side-by-side on a page to indicate their relationship.
- Provide adequate space between the fields.
- Do not overlap them; otherwise, they appear skewed when viewed through the browser.
- Change the label type for the related field to *None* so that it doesn't appear on the page.
- Enter a text description to document the purpose of the field.

**Note.** The display control and related fields must be in the correct order on the order page. The display control field must be positioned before the associated related field; otherwise, a warning dialog box appears when you save the page. Also, fetching the related field after the control field is entered requires a transmission to the server at runtime.

## Creating Related Edit Fields

Related edit fields are slightly different from related display fields. Related edits enable users to enter descriptive values instead of code values and enable prompting on those descriptive values. Changing the value in the control field updates the related edit field, and changing the value in the related edit field updates the value in the control field. If appropriate, the control field can be display-only or even invisible to hide code values from the user.

At runtime, when a user edits a related edit field, it is treated as if the user changed the control field directly. Edits, including any assigned PeopleCode, are run on the control field. PeopleCode is not run for the related edit field.

The following partial screen shot is an example of a related edit field with a control field that is available for entry. When entering company information, the data-entry clerk may not have all company codes memorized, but instead knows just the name of the company. By entering the more descriptive company name in the Description field, (the related edit field) the company code (display control field) is automatically populated in the Company field.

If the user enters *Payroll* in the Description field and exits that field, the system searches for a match to this value and populates the field if only one is found. The field reads *Payroll Services Technology*. If more than one match is found, the field turns red, and an error message appears indicating that the user must use the lookup page to view all related values. Alternatively, the user could enter more detailed information in the field before clicking the prompt to further narrow the search.

Related edit field

At design time, you set up a related edit field in the same way that you create a related display field, except that you clear the Display Only check box in the Field Use Options group box so that the field is active at runtime. The control field's prompt table is used. You do not need to define a prompt table for the related edit field, just the control field. The related edit field must be a key or an alternate key in the prompt table. PeopleSoft recommends that the related values in your prompt table be unique so that there is a one-to-one mapping to the control values. This prevents the user from receiving a warning and having to access the prompt page to select the proper value.

In some instances, you may not want the control field to be apparent to the user. PeopleSoft Global Payroll, for example, relies heavily on PINs. While PINs are key values, and therefore very important for tracking and storing information in the database, you may not want to distract the user with such a number. However, you still want this data to be entered into the database with the transaction. Related edit fields are particularly helpful in these situations. You can associate the PIN field with a descriptive field and have only the descriptive, related edit field appear on your page. To do so, hide the control field using PeopleCode.

---

**Note.** If the hidden control field that you select is a required field, PeopleSoft Pure Internet Architecture changes the related edit field into a required field as well. At runtime, an asterisk appears next to the field label, marking the field as required. If the user does not enter information in the related edit field, the system displays an error message.

---

### See Also

*PeopleTools 8.52: PeopleCode Language Reference, "PeopleCode Built-in Functions," Hide*

## Creating Invisible Control Fields

Your page design may require an invisible control field. For example, if users are interested only in the contents of the related display field, then make the control field invisible. Unlike visible control fields, you *can* overlap an invisible control field and its related display field.

---

**Note.** If you specify a field as invisible in PeopleSoft Application Designer, you cannot make it visible using the UnHide property in PeopleCode.

---

To add an invisible control field with a visible label:

1. Access the properties dialog box for the invisible control for which you want the label to appear.
2. Select the Use tab.
3. Set the appropriate field use options:
  - a. Select Display Only, Invisible, and Show Label to make the field invisible and its label visible.
  - b. Select Display Control Field.
4. Select OK.
5. Add an edit box to your page.

Place this edit box directly to the right of the display control field that you just set.
6. Access the properties dialog box for the new edit box.
7. Select the Use tab.
8. Select Related Field.
9. Select the appropriate control field from the Related Control Field drop-down list box.
10. Select OK.

---

## Ordering and Spacing Page Controls

This section provides an overview of page control order and spacing and discusses how to:

- Order controls visually.
- Order controls logically.
- Test page control order.
- Apply control order rules.
- Change control order using the order list.
- Find a field on a page definition.
- Use default order.
- Space controls on pages.

## Understanding the Page Definition Order Tab

Page definitions include an **Order** tab. Here, the system populates a grid with the names and properties of the page controls associated with the page definition. Properties, which appear as grid column headings, include:

- Tab Order

- Field ID
- Level
- Label
- (Control) Type
- Field
- Record
- Display Control
- Related Field
- Control Field
- Page Field Name
- Allowed Deferred Processing
- Popup Help

You can use the Order tab as a diagnostic tool, which can save time in addressing issues your page might have with page field processing order and page field tab order among other visual and logical page-processing properties. All values on the Order tab are display-only. Some property values, such as record and field names appear as text; other values, such as Popup Help appear as check boxes in which case, a check indicates that the property is applicable to the page control.

## Understanding Page Control Order and Spacing

The order of the page controls that are on your new page is important to both how your users interact with the page and how the component processor interprets the underlying record and field relationships.

Two types of page control order are important on pages:

- The order in which you visually arrange controls on the page.
- The logical processing order—governed by levels—that the system requires to correctly process the page data.

As you add controls to a page, the system automatically builds a processing control order list that can be viewed on the Order tab of the page definition. Usually, you don't want to preserve this order for processing, so you can reorder your controls on the Order tab to indicate how the system should process your page.

## Ordering Controls Visually

Organize fields from top left to bottom right in a page. Include most important information at top. Use group boxes to group related fields and indicate the hierarchy of information on a page. Guidelines for grouping information include:

- Keys

Group all key fields at the top of a page (such as Business Unit, Employee ID, and SetID).

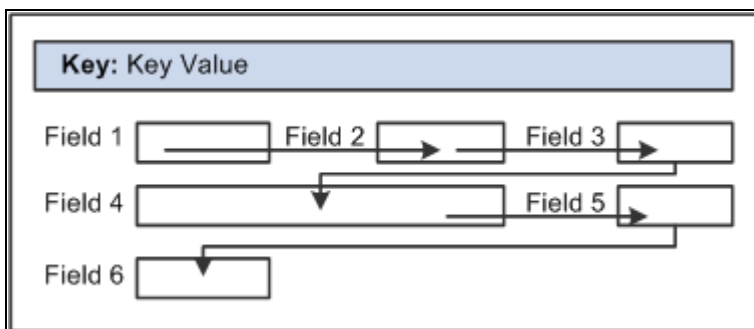
- Level 1-3

Controls at these levels must be enclosed by a scroll area or grid. At design time, these levels can be stacked. At runtime, they appear nested.

## Ordering Controls Logically

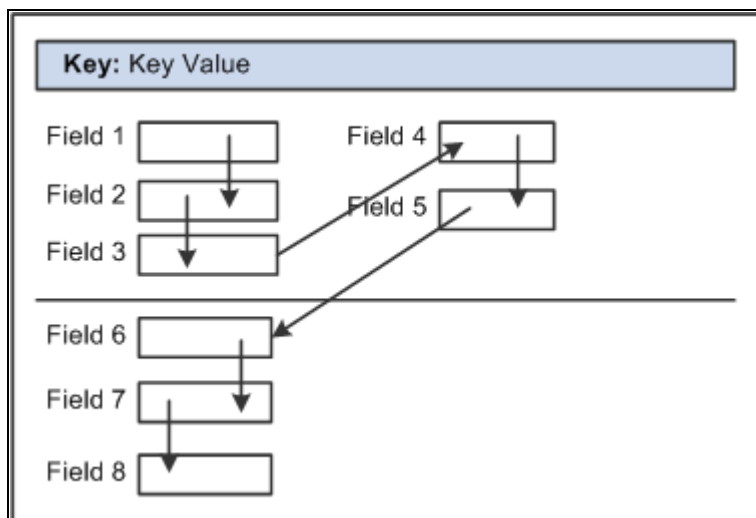
Tab ordering through a page at runtime is strictly defined by page field ordering. When you add a control to a page, PeopleSoft Application Designer automatically inserts it in the order list based on where you position it on the page. You can manipulate the ordering of fields on your page by moving them up or down on the Order tab in the page definition. In doing so, you also change the tab order that users follow as they move through fields on the page.

In general, field tab order should flow from top left to bottom right on a page. The flow should reflect the way that the eye naturally scans information. On a page with multiple scroll levels, the tab order should follow the scroll level order, from Level 0 to Level 1, and so on. In a noncolumnar page, the tab order should go from left to right on each line from top to bottom as depicted in this figure:



Example flow of noncolumnar page

In a columnar page, the tab order should go from the top to the bottom of the first column, then from the top to the bottom of the second column, and so on.



Example flow of columnar page

Fields that share the same label should follow consecutively in the tab order from left to right.

## Testing Page Control Order

After you have placed all of the controls on your page, you should test the tab order. You can do this using either the test mode or by viewing the page in the browser. Using the View Page in Browser mode is more useful because you can also check the placement of controls and subfields that are on your page. In these testing modes, you can enter data into edit boxes and select radio buttons and check boxes. However, you cannot save data that you enter, and push buttons and links do not function.

---

**Note.** The tab order that you establish at design-time may not apply for all browsers.

---

See *Troubleshooting Browser Limitations* available on the My Oracle Support website. [support.oracle.com](http://support.oracle.com).

To test tab flow using View Page in Browser mode:

1. Open the page in PeopleSoft Application Designer that you want to test.
2. Select Layout, View in Browser.
3. Select the appropriate browser from the drop-down list box.
4. Press the Tab key to move from one field to the next.

Press Shift+Tab to move to the previous field.

## Applying Control Order Rules

If you're working with a complex page or you're new to order lists, you might print out a page definition and work on the hard copy. The sequence number of each control is reflected in the Num column on the page report. You can easily mark where you need to move a control to make your page function properly, or you can look at the Lvl column on the Order tab of the page definition.

Use these rules to order your controls:

### ***Radio Buttons***

For radio buttons to function in a single group, they must be associated with the same record field and be listed together on the Order tab of the page definition. The only control that you can place between related radio buttons is the static text control to extend radio button labels. Put the text immediately after the radio button to which it relates.

### ***Level-Based Controls***

List level-based controls (scroll areas, grids, and scroll bars) immediately before the first control that they govern, followed by the controls that are directly governed by that control. Level-based controls directly govern all controls that are listed below them on the order list until they encounter another level-based control that is at the same or lower level (higher occurs level number).

### ***Display Controls***

Place display controls before related displays that they govern. The related display controls don't have to follow the display controls immediately, but they must be inside the same scroll area or scroll. However, if you have more than one related display control, placing each immediately following its display control makes the order page easier to read and understand.

## **Changing Control Order Using the Order Tab**

The Order tab of the page definition displays attributes about each of the page controls and their processing order. The Field ID column represents the order in which the control was added to the page. If you modify the order of page fields on the page, note that the numbers assigned to each field remain constant. Thus the IDs might not always appear in sequential order. When a database compare is performed, the Field ID appears on the Compare report to assist you in identifying specific page fields.

Reorder page fields on the Order tab by dragging them in the same view. Changing the order list doesn't change the physical location of controls on the page. It changes only the logical order or tab order in which controls are processed. When you have added and arranged all of your controls, you may want to print your page definition to see how you might need to reorder your controls.

The Order tab also governs processing rules for scrolls and record or field relationships. Consider which controls are associated with which scroll area or scroll bar and which secondary relationships are important to page processing.

To change the order list:

1. Open the page.
2. Select the Order tab on the open page.

The Order view appears. Use this view to change the logical processing order of fields on your page.

3. To move a control to another position in the control order list, select the control that you want to move.

Press the Shift key to select multiple controls.

4. Drag the selected control to the new position on the Order view.

The system moves the control to the new location in the list. The ID value remains static. The visual display of the page still looks the same—changing the order list doesn't move the control on the page, only the logical processing order of the control.

---

**Important!** When viewing your page in "Test Mode," page fields cannot be accessed, moved, or modified from the Order tab if the page definition contains a sub page. Exit "Test Mode" to rearrange page fields in the order list.

---

## Finding a Field on a Page Definition

Use the Order view to locate a field on a page definition. Using this method of locating fields is useful when you are working with detailed page definitions that contain many fields.

To find a field on a page:

1. Access the Order tab of your page definition.
2. Select the field that you want to locate on the page.
3. Select the Layout tab.

The field that you selected on the Order tab is selected in the Layout view.

## Using Default Order

You can reorder the field list on the Order tab so that the logical control order reflects the physical control order as shown on the Layout tab. Do so by selecting Layout, Default Page Field Ordering. You can use this tool in either the Layout or Order view for your page definition. The system creates the default order by reading the page as though it was on a pixel-by-pixel grid, from top to bottom and left to right.

---

**Note.** Use the default ordering option sparingly on existing pages. It may offset any previous order that was created. However, on new pages, you may find default ordering a useful starting point for your control order list.

---

## Spacing Controls on Pages

Provide adequate space between controls on a page so that fields and their labels don't touch or overlap other controls or labels in PeopleSoft Application Designer. If they do, then when your page appears in the browser, the label is offset automatically and the overlapped control might shrink.

### ***Overlapping Fields***

You can layer or overlap fields only if the one that you are overlapping is set to *Invisible* on the Use tab of the properties dialog box for that control. When you move through the data entry controls on a page, the component processor recognizes only visible, unhidden fields as available for entry.



You can layer multiple invisible fields. However, you must set up your stacks of edit boxes so that all but one field in the stack are invisible or hidden at RowInit time.

## Designing Inquiry Pages

You can design pages for inquiry purposes only. Inquiry pages are usually based on search records that extract specific information from different tables to display different views of your database, such as a summary of expenses by department.

To create an effective inquiry page:

- Make all of the fields display-only.
- Build in sufficient display control and related field relationships to show relevant descriptions.
- When using level-based controls, disable the row action buttons so that the user cannot add or delete rows.

Workforce Job Summary inquiry page

## Aligning Page Controls and Page Control Labels

When you are in page design mode, the Align and Align Label buttons on the page definition toolbar enable you to align your page controls and their labels relative to each other, both horizontally and vertically.

PeopleSoft Application Designer uses relative alignment; the *last* item that you select is the guide, or anchor. All selected page controls or page control labels align relative to the anchor. The anchor does not move when you select an alignment option.

You can align page controls and their labels together by using the Align buttons. Both the control and its label move when you use this option. You can also align labels by themselves without moving the associated controls by using the Align Label buttons. The Align and Align Label buttons are enabled when you select two or more page controls or page control labels.

In addition to using the Align buttons, from the menu you can also select Layout, Align and then the control or label alignment position.

**Note.** Align the page controls *before* you align their labels.

## ***Aligning Page Controls***

To align page controls:

1. Select the controls that you want to align by pressing Ctrl and then clicking each page control.

---

**Note.** You can also select multiple page fields by clicking and dragging the cursor around the controls. However, for relative alignment purposes, the Ctrl + click is the preferred method of selection, because it enables you to explicitly set the anchor control.

---

2. Click the appropriate Align button in the toolbar: Align Left, Center, Right, Top, Middle, or Bottom.

When the page controls move, their labels move with them maintaining relative positioning to the associated controls.

You can manually move page controls after relative alignment, though previous alignment is not preserved.

## ***Aligning Page Control Labels***

To align page control labels:

1. Select the labels that you want to align by pressing Ctrl and then clicking each label *or* control.
2. Click the appropriate relative alignment button in the toolbar: Align Label Left, Center, or Right.

When the labels move, their controls remain stationary.

You can manually move labels after relative alignment, though previous alignment is not preserved.

## ***See Also***

Chapter 11, "Creating Page Definitions," Selecting Controls, page 332

---

# **Maximizing Performance**

Page definitions permit controlled access to application data. The system can validate the data, write it to the database, and then retrieve and display it upon request. Behind the scenes, the component processor—the PeopleTools runtime processor—builds SQL statements that are based on the actions that you perform on pages. The component processor:

- Manages the flow of data processing as users enter information on pages.
- Issues INSERT, DELETE, and UPDATE statements to maintain data in the database and SELECT statements to retrieve data.

As you design your pages, some features can adversely affect page performance. There's always a tradeoff between eliminating a useful feature and speeding up page processing. Guidelines for improving page performance include:

- Be judicious about references to record definitions other than the primary record definition under each scroll area.

References to other record definitions can include:

- Related display controls.
  - PeopleCode references (such as edits and defaults) to other records.
  - Defaults to fields on other record definitions.
  - Field controls on derived or work records.
- Put the field control on the appropriate derived or work record, rather than on a regular data record definition to derive its value.

For example, FTE (full-time equivalent) on the JOB record definition would be moved to the DERIVED\_HR derived or work record because its value is derived by the system.

- Use as few record definitions as possible in a component.

When you open a page in a component, the system loads all record buffers from the entire component into buffers.

- Do not remove table edits to improve performance, even though an edit against another table causes a short pause.

Eliminating them may compromise data integrity.

- Frames, scroll areas, scroll bars, grids, and group boxes all create HTML tables, which, in some browsers, might slightly degrade the performance of your system.

Be judicious about the number of these types of controls that you use on a page.

- Apply deferred processing when possible to fields, pages, and components.

### **See Also**

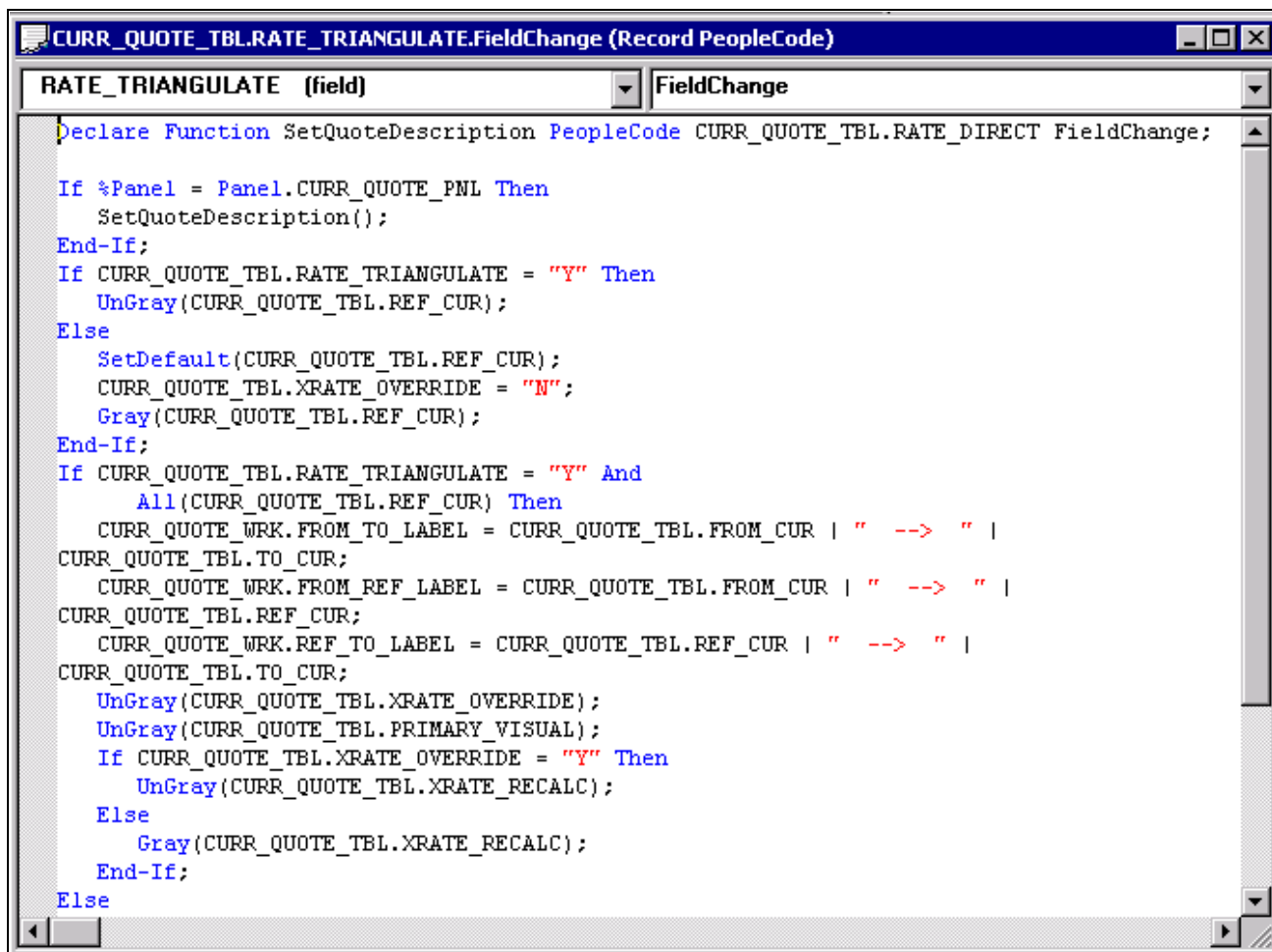
Chapter 8, "Designing Pages for Optimal Performance," page 199

*PeopleTools 8.52: PeopleCode Developer's Guide, "PeopleCode and the Component Processor"*

---

## **Accessing PeopleCode in Page Definitions**

Page fields are associated with a specific record field. You can access the PeopleCode for that record field from the page field in the page definition. You might add PeopleCode to a field to achieve a variety of tasks, ranging from real-time editing of fields to altering the appearance of fields on a page. You can also associate PeopleCode with a component, component record, and component record field.



### Editing PeopleCode

To edit or add record field PeopleCode through a page definition:

1. Select File, Open to open the page definition through which you want to access PeopleCode.
2. Select the field on the page for which you want to view the associated PeopleCode.
3. Select View, View Record PeopleCode.

The PeopleCode editor appears, providing access to all of the PeopleCode for the record that owns that field.

4. Select the PeopleCode event type from the drop-down list box in the PeopleCode editor.

You can also select other fields in the primary record from that drop-down list box.

5. Select File, Save to save your changes.

### See Also

*PeopleTools 8.52: PeopleCode Developer's Guide*, "Accessing PeopleCode and Events"

---

## Viewing Pages in the Browser

This section discusses how to:

- Work with the View in Browser feature.
- Change the default browser.
- Generate HTML.
- View the HTML for a page.
- Identify page definitions online.

### Working With the View in Browser Feature

During the page design process, you should periodically view how your page looks in the browser. How you design your page in PeopleSoft Application Designer may look different when viewed online, particularly if you are using subpages that contain multiple fields.

To do this, select Layout, View in Browser. This feature is helpful if you are changing any of the style characteristics of the page or aligning and spacing controls, such as scroll areas.

This is the PSOPTIONS page definition:

**PSOPTIONS (Page)**

**Layout** | **Order**

**PeopleTools Options**

Environment Long Name:  Environment Short Name:

System Type:

**Language Settings**

Language Code:  Sort Order Option:

☐ Translations Change Last Update

Background Disconnect Interval:

☐ Multi-Company Organization

☐ Multi-Currency

☐ Use Business Unit in nVision

☐ Use Secure Rep Rqst in nVision

☐ Multiple Jobs Allowed

☐ Allow DB Optimizer Trace

☐ Grant Access

☐ Platform Compatibility Mode

☐ Allow NT batch when CCSID<>37

☐ Save Error is Fatal

☐ Set Focus on Save Button

Temp Table Instances (Total):

Temp Table Instances (Online):  22

Maximum Message Size:

Base Time Zone:

Last Help Context # Used:

Data Field Length Checking:

Maximum Attachment Chunk Size:

Upgrade Project Commit Limit:

Enable Switch User:

Sample page in PeopleSoft Application Designer

This is the same definition in View In Browser mode.

Same sample page viewed in Microsoft Internet Explorer

The View in Browser feature provides only a rough idea of how your page might look. For a more complete view of the page at runtime, including the navigation header, assign your page to a component and add it to the existing menu structure using the Registration Wizard. You can then access the page from the menu in the PeopleSoft Pure Internet Architecture.

## Changing the Default Browser

You can test your page in any browser by changing the default browser to one other than Microsoft Internet Explorer. After you set up a new browser, you can select in which browser you want to view your pages.

To change the default browser:

1. Open a page in PeopleSoft Application Designer.
2. Select Layout, View in Browser, Edit Browser List.

The Browsers dialog box appears.

3. Click the Add button.
4. Click the Browse button to locate the browser to add.

You must select the correct path and browser application for the new browser to function properly.

5. Enter the name, type, and version of the browser.

The name that you enter appears in the Layout, View in Browser menu.

## Generating HTML

The PeopleSoft Pure Internet Architecture system automatically writes HTML forms or tags for all of the page controls that you add to a page. You do not need to know how to write HTML code to create pages in PeopleSoft Application Designer and then view them in the browser. The table shows this conversion:

### *Page Controls as HTML*

<i>Page Controls</i>	<i>HTML Forms</i>	<i>Static HTML Tags</i>
Edit Box (editable)	<INPUT TYPE=TEXT>	N/A
Long Edit Box	<TEXTAREA>	N/A
Drop-down List Box	<SELECT><OPTION><OPTION>...<=> /SELECT>	N/A
Radio Button	<INPUT TYPE=RADIO>	N/A
Check Box	<INPUT TYPE=CHECKBOX>	N/A
Push Button or Link	<INPUT TYPE=BUTTON> <INPUT TYPE=HYPERLINK>	N/A
Static Text	N/A	HTML text
Static Image	N/A	<IMG>
Grid	N/A	HTML <TABLE>

You can view the HTML that is created for your page by using the Generate HTML feature. The Generate HTML feature writes the HTML code to a file on the local drive. This feature is mainly used for debugging purposes and is not needed for the creation of pages.

To generate HTML for a page:

1. Open the page for which you want to generate the HTML.
2. Select Layout, Generate HTML.



3. Select the browser type and version.

PeopleSoft Application Designer generates the HTML to the c:\TEMP\pshtml directory. The file is named according to the page name and the browser that you select. For example, the Absence History page is named ABSENCE\_HISTORY IE5.00.html.

## Viewing the HTML for a Page

To view the HTML for your page:

1. Open the generated HTML file.

The selected browser appears, displaying how your page appears at runtime.

2. Select View, Source in your browser window.

Notepad or another text-based application appears with the HTML that was used to create your page.

---

**Note.** Different browsers might have different methods for viewing the HTML for your page.

To adjust the look or function of how your page looks in the browser, do so in PeopleSoft Application Designer. If you make changes directly to the HTML in Notepad, your changes are *not* saved to the system. Each time that you open the page in the browser, it reverts to the settings of the original page definition.

---

## Identifying Page Definitions Online

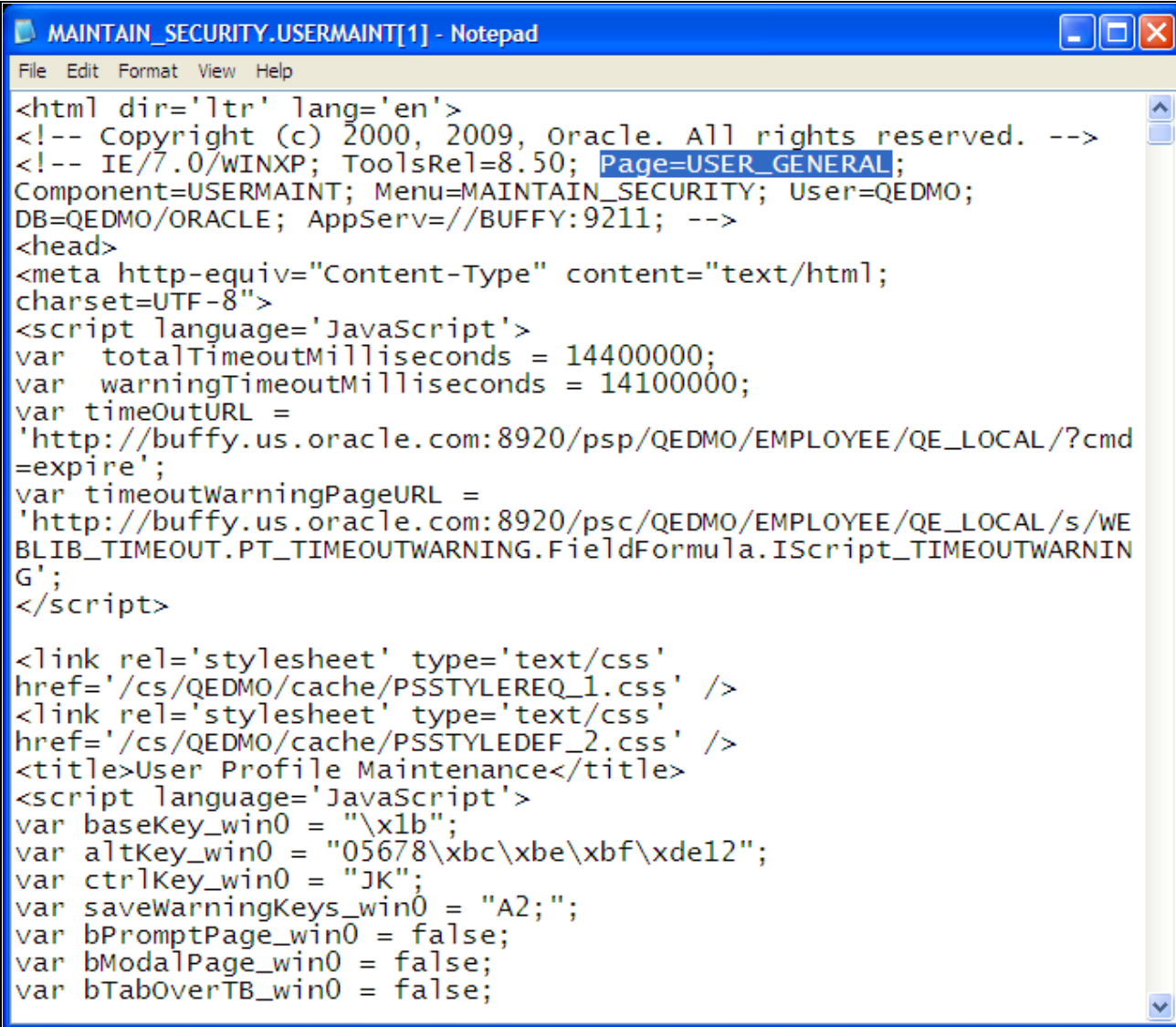
In many cases, the page name online differs from the actual name of the page definition in PeopleSoft Application Designer. While viewing a page in the browser, locate the page definition name by pressing Ctrl + J. A browser page appears revealing the name of the page, along with other system elements:

Browser	IE/7.0
Operating System	WINXP
Browser Compression	ON (gzip)
Tools Release	8.50-808-R1
Application Release	PeopleTools 8.50.00.000
Service Pack	0
Page	USER_GENERAL
Component	USERMAINT
Menu	MAINTAIN_SECURITY
User ID	QEDMO
Database Name	QEDMO
Database Type	ORACLE
Application Server	//BUFFY:9211
Component Buffer Size (KB)	62

[continue](#)

System elements displayed when pressing Ctrl + J

You can also right-click the page and selecting *View Source* from the drop-down menu. The HTML code appears for your page in a text application, such as Notepad. In most cases, the page name is in the sixth or so line of code after *Page:*. You can also identify the component and menu definitions for your page next to the page definition name.



```

MAINTAIN_SECURITY.USERMAINT[1] - Notepad
File Edit Format View Help
<html dir='ltr' lang='en'>
<!-- Copyright (c) 2000, 2009, Oracle. All rights reserved. -->
<!-- IE/7.0/WINXP; ToolsRel=8.50; Page=USER_GENERAL;
Component=USERMAINT; Menu=MAINTAIN_SECURITY; User=QEDMO;
DB=QEDMO/ORACLE; AppServ=//BUFFY:9211; -->
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
<script language='JavaScript'>
var totalTimeoutMilliseconds = 14400000;
var warningTimeoutMilliseconds = 14100000;
var timeoutURL =
'http://buffy.us.oracle.com:8920/psp/QEDMO/EMPLOYEE/QE_LOCAL/?cmd
=expire';
var timeoutWarningPageURL =
'http://buffy.us.oracle.com:8920/psc/QEDMO/EMPLOYEE/QE_LOCAL/s/WE
BLIB_TIMEOUT.PT_TIMEOUTWARNING.FieldFormula.IScript_TIMEOUTWARNIN
G';
</script>

<link rel='stylesheet' type='text/css'
href='/cs/QEDMO/cache/PSSTYLEREQ_1.css' />
<link rel='stylesheet' type='text/css'
href='/cs/QEDMO/cache/PSSTYLEDEF_2.css' />
<title>User Profile Maintenance</title>
<script language='JavaScript'>
var baseKey_win0 = "\x1b";
var altKey_win0 = "05678\xbc\xbe\xbf\xde12";
var ctrlKey_win0 = "JK";
var saveWarningKeys_win0 = "A2;";
var bPromptPage_win0 = false;
var bModalPage_win0 = false;
var bTabOverTB_win0 = false;

```

Page name in HTML source code

## Producing Pages

This section provides an overview of page production and discusses how to:

- Change general page information.
- Change page use information.
- Save pages.
- Rename and delete pages.
- Print page definitions.
- Alter page setup.

- Read a page definition report.

## Understanding Page Production

If you change pages or delete controls in the page definition, use the tools in the Edit and File menus. To avoid making changes that might adversely affect your application database, review your plans with your database administrator. Together, you can evaluate the effect that your actions might have on the system database as a whole.

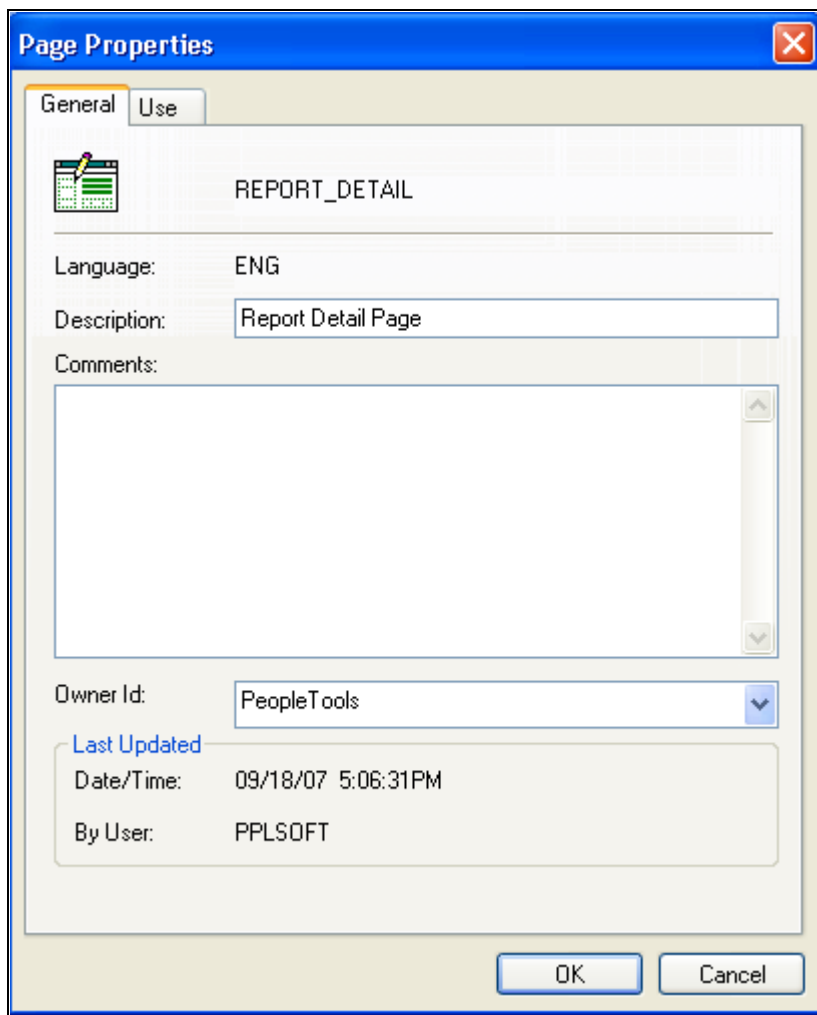
Several additional steps in the page design process help make future editing of page definitions easier.

After you design a page, you must set the attributes for how the page should function and make sure that you have the proper documentation to assist you when performing upgrades. Access the Page Properties dialog box by selecting File, Definition Properties in page definition mode.

Use the General tab in the Page Properties dialog box to document the page. Enter both short and long descriptions of the page here. You can also document what changes have been made, or must be made, to a page.

## Changing General Page Information

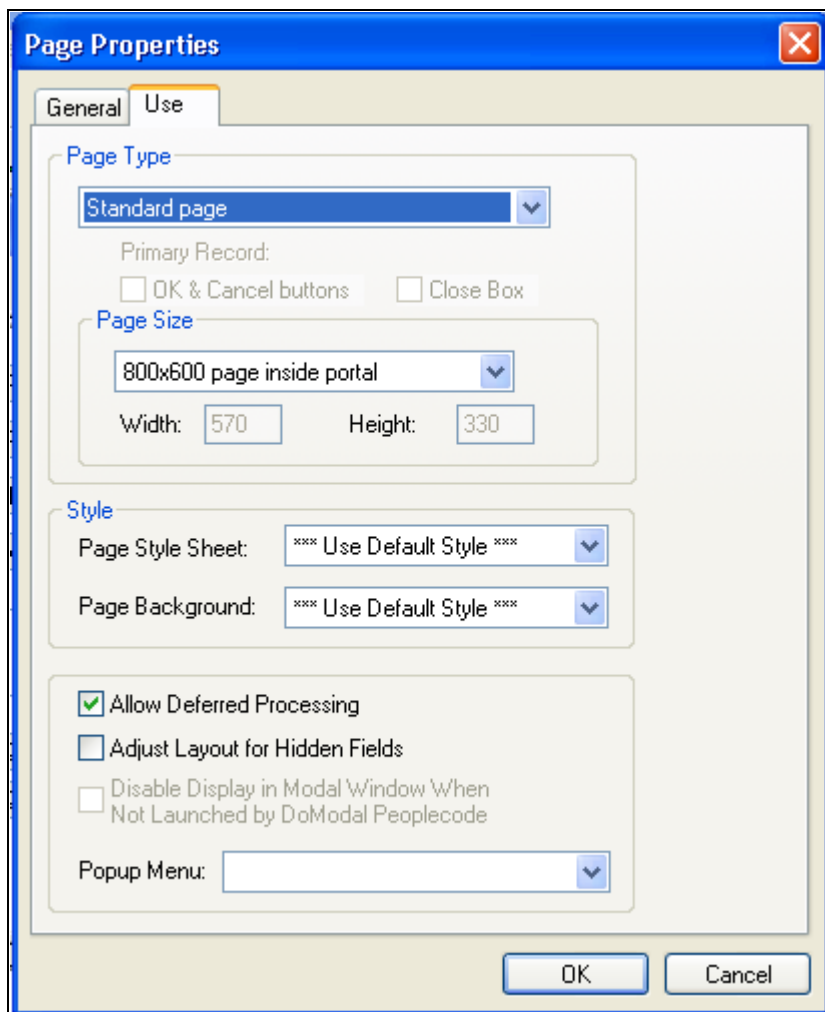
Access the Page Properties dialog box and select the General tab. Enter a description and any comments about the page and assign an Owner ID.



Page Properties dialog box: General tab

## Changing Page Use Information

Access the Page Properties dialog box and select the Use tab.



Page Properties dialog box: Use tab

### Page Type

Select *Standard page*, *Subpage*, *Secondary page*, or *Popup page* from the drop-down list box.

See [Chapter 9, "Using Page Controls," Understanding Subpages, page 237.](#)

See [Chapter 9, "Using Page Controls," Understanding Secondary Pages, page 239.](#)

See [Chapter 9, "Using Page Controls," Using Pop-up Pages, page 226.](#)

### Page Size

To accommodate different types of workstation monitors, you can adjust the page size.

#### 800×600 page inside portal

This value is the standard option for viewing pages in the portal, providing room for the universal navigation header and the menu. It provides a default page size of 570×330 in PeopleSoft Application Designer.

<b>800×600 page without portal</b>	You should develop most pages using this option. It is the standard option for viewing pages, providing room for the navigation header and the menu. This option provides a default page size of 760×330 in PeopleSoft Application Designer.
<b>1024×768 page inside portal</b>	Selecting this option provides an optimal display for users with a 1024×768 resolution, showing the full transaction page and the menu. Users with an 800×600 display will see the full transaction page without the menu when the menu is enabled. This option provides a default page size of 760×498 in PeopleSoft Application Designer.
<b>1024×768 page without portal</b>	For the power user, this is a high-resolution option. It provides a default page size of 984×498 in PeopleSoft Application Designer.
<b>240×Var portal home page comp.</b> (240×variable portal home page component)	Use this setting only for creating pagelets for the portal. This size is most usable because it can be viewed in both the narrow area and the wide area of the portal. The system sets the width to 240, but you set the height.
<b>490×Var portal home page comp.</b> (490×variable portal home page component)	Use this setting only for creating pagelets for the portal. Pagelets that use this setting can appear only in the wide area of the portal. The system sets the width to 490, but you set the height.
<b>640×480 Windows screen</b>	This option is a deprecated feature used only for Windows client users. This option is designed for VGA resolution. The pages are actually less than 640×480 pixels in size in PeopleSoft Application Designer because they provide space for various windows items, such as the window title, menu bar, and toolbar.
<b>800×600 Windows screen</b>	This option is a deprecated feature used only for Windows client users. This option is designed for Super VGA resolution. These pages provide space for the window title, menu bar, toolbar, folder tabs, and status line, as well as space at the bottom for the taskbar.
<b>Custom size</b>	Select to set a specific page size other than those listed previously. Set the width and height manually.

## Style

<b>Page Style Sheet</b>	Select a different style sheet for a specific page to override the style sheet that is selected for the application. If you do not select a different page style sheet (keeping <i>***Use Default Style***</i> ), the system uses the style sheet that is specified on the PeopleTools Options page.
<b>Page Background</b>	Select a different page background style class to override the background style of the page style sheet specified above. If you keep the <i>***Use Default Style***</i> setting, the system uses the default background of the page style sheet.

---

**Note.** The style options that you select in the design time environment might look different at the runtime rendering of the page. For example, font and page control sizing metrics are different in the design time Windows environment and the runtime browser environment. As such, when developing pages, do not rely entirely on the rendering of the page in PeopleSoft Application Designer to gauge layout considerations, such as font size and the placement of controls. You should view all pages in the runtime browser environment and adjust them accordingly.

---

### ***Additional Settings***

<b>Adjust Layout for Hidden Fields</b>	Select to set the page to resize automatically when hidden fields are present.
<b>Pop-Up Menu</b>	See <a href="#">Chapter 13, "Creating Menu Definitions," Defining Pop-up Menus, page 429.</a>
<b>Allow Deferred Processing</b>	Deselect if you want the page and all of its fields to follow standard processing. Deferred processing is the default.

### ***Modal Window***

These options are available when you select *Secondary page* as the page type.

<b>None</b>	Select to have the secondary page appear in the target area of the browser, similar to standard transaction pages.
<b>Display in Modal Window</b>	Select to have the secondary page appear in its own window, hovering over the transaction page, which is grayed out until you dismiss the secondary page.

## **Saving Pages**

To save your work, select File, Save or File, Save As.

When you save your page, the system performs various edits and issues warning messages as appropriate. The warning explains any errors that are associated with each control. You can disregard these messages and save your work anyway, but you cannot view the page in the browser until all errors are corrected.

Access the explanation of an error message by clicking the Explain button in the warning message box. You can fix the problem immediately or save the page and return later to correct it.

## **Renaming and Deleting Pages**

If you rename a page, the change is reflected throughout the system, including in PeopleCode, so you do not have to change the page name anywhere else.

To delete a page, particularly a subpage or a secondary page, first use the Find Definition References option to determine which page, component, and menu definitions refer to or use the page you want to delete.



**See Also**

[Chapter 2, "Using PeopleSoft Application Designer," Renaming Definitions, page 34](#)

[Chapter 2, "Using PeopleSoft Application Designer," Deleting Definitions, page 35](#)

[Chapter 2, "Using PeopleSoft Application Designer," Finding Definitions, page 36](#)

**Printing Page Definitions**

To keep track of your application definitions and refinements, print your page definitions and keep a log or binder with your new and revised pages as you reconfigure your applications. You may want a reference for each page that includes both the onscreen image and the page control information (the record definitions and special uses or edits that you applied), as well as the information for any subpages that are included in the page definition.

Printing page definitions also helps you debug during testing. Each report includes a unique identifier that the system automatically increments each time you change the page.

To print a page definition:

1. Open the appropriate page.
2. Select File, Print.

**Altering Page Setup**

Access the Print Setup dialog box by selecting File, Print Setup.

***Print***

All print options are selected by default.

<b>Definition</b>	Select to print the list of fields on the page and all of the parameters that you set for each one.
<b>PeopleCode</b>	Select if you used PeopleCode to design your page
<b>Graphics</b>	Select to provide a snapshot of your page design. Graphics generally take longer to print than page descriptions or PeopleCode reports.

***Borders***

<b>Header</b>	Select to print a header at the top of the report indicating the date and time that you printed the report and the database name, page name, version number, and page number. This option is selected by default.
---------------	---

<b>Footer</b>	Select to print a footer at the bottom of the report indicating the date and time that you printed the report and the database name, page name, version number, and page number. This option is deselected by default.
<b>Border</b>	Select to print a border or box around a page report. This option is deselected by default.
<b>Border Space (mm)</b> (border space (in millimeters))	Select to insert a set amount of space between a graphical border around the page report and the margins of the report. Deselect this option to print in character mode (faster) rather than in graphics mode (slower). The default space is 3 mm, but you can change it.

### ***Margins***

Set the distance, in millimeters, from the edge of the page to the left, right, top, and bottom edges of the page image or report. The defaults are:

- Top: 20.
- Bottom: 5.
- Left: 10.
- Right: 0.

---

**Note.** The HP LaserJet printer is the default.

---

## **Reading a Page Definition Report**

This table lists the columns on the page definition report and describes their content:

<b><i>Column</i></b>	<b><i>Content</i></b>
Num (Number)	Sequential number that shows the relative position of the field in the field order list for the page.
Field Type	Field type, such as edit box.
Label Text	The label associated with the field, regardless of whether the label appears on the page.

<b>Column</b>	<b>Content</b>
Label Type	<ul style="list-style-type: none"> <li>• None: Label does not appear on the page.</li> <li>• Text: Text label.</li> <li>• RFT Long: Long name for the field (from the record definition).</li> <li>• RFT Short: Short name for the field (from the record definition).</li> <li>• XLAT Long: Long name for the value (from the Translate Table).</li> <li>• XLAT Short: Short name for the value (from the Translate Table).</li> <li>• Bitmap: Bitmap appears on button face.</li> </ul>
Record Name	Record name where the field is located.
Field Name	Field name on the record definition with which the page field is associated.
Siz (size)	Field size (custom, minimum, or maximum).
Alg (alignment)	Alignment (left or right).
On	Value of the radio button or the on value of the check box (typically Y).
Off	The value of the radio button or the off value of the check box (typically N).
DSP (display)	Yes indicates a display-only field.
INV (invisible)	Yes indicates an invisible field.
CTL (control)	Yes indicates a display control field.
REL (related)	Yes indicates a related display field.
RelNum (related number)	Number (Num column) of the display control field for this related display field.
CUR (currency)	Yes indicates a multicurrency field that does not appear unless the user is operating in multicurrency mode.
DER (derived)	Yes indicates a field from a derived or work record.

<b>Column</b>	<b>Content</b>
OccLvl (occurs level)	Occurs level of scroll bar that governs this field.

You may see additional reference lines following each field or at the end of the report:

<b>Reference</b>	<b>Description</b>
Page Help Context: <i>nnn</i>	Indicates that the page was assigned a page help context number that links it to a help file that describes how the field is used wherever it appears in the database.
Field Help Context: <i>nnn</i>	Indicates that this field, as used on this page, was assigned a record field help context number that links it to a help file that describes how the field is used only as it appears in this record definition.

---

## Working with Pages in Foreign Languages

This section provides an overview of working with pages in a foreign language and discusses how to:

- Perform a Save As using foreign language pages.
- Delete foreign language pages.

### Understanding Working with Pages in a Foreign Language

Many organizations are multinational, operating their business in multiple countries and continents. For efficiency, these organizations typically deploy a single application in multiple languages. However, because internal application developers and administrators may not be familiar with each language in which applications are deployed, this can cause difficulty when updating or maintaining the applications. To assist developers at multinational organizations, PeopleSoft provides a Save As option and a Delete option when working with foreign language pages.

### Performing a Save As Using Foreign Language Pages

PeopleSoft Application Designer enables developers to modify pages in their base language and save the page in another language. This capability decreases the time required to develop pages in other languages, and it relieves developers from having to modify pages in languages they do not understand. To open a page definition in your base language and save the changes to another language, use the Save As option.

The Language drop-down list box on the Save As dialog box is only available when you are signed onto the database in the base language. When performing the Save As, select the appropriate language from the Language drop-down list box.

If the page definition exists in the foreign language (in the nonbase language), the system prompts you to confirm an overwrite of the definition.

---

**Note.** You can save modifications to a nonbase language only if the base language definition already exists. If the base language page does not exist, then the system saves the page to the base language, regardless of the language selected.

---

---

**Note.** After doing a Save As to another language, the page will still appear in PeopleSoft Application Designer in the base language.

---

## Deleting Foreign Language Pages

When you are signed onto the database in the base language, the Delete dialog box contains the Language drop-down list box. The Language drop-down list box enables you to delete multiple pages from multiple nonbase languages.

---

**Note.** Deleting the base language page automatically deletes all translated pages for the deleted base language page.

---



## Chapter 12

# Creating Component Definitions

This chapter provides an overview of component definitions and discusses how to:

- Define components.
- Specify market-specific components.
- Determine where components are used.
- Set component properties.
- Use search records.
- Access the Message Catalog.
- Enable the Expert Entry feature in the Security component.

---

## Understanding Component Definitions

A component represents a complete business transaction. It comprises either a single page or a set of pages that is meant to be processed as one page. After you create pages, you add them to one or more components to access them through menus or in business processes.

This section discusses:

- What component definitions manage.
- The component buffer.

## What Component Definitions Manage

Component definitions manage:

- Groupings of pages and their associated tab labels.
- Search records that are used to retrieve data for the page.
- Access keys for folder tab navigation.
- Links at the bottom of a page.
- The toolbar at the bottom of a page.

This example shows a sample component with multiple pages:

The screenshot displays a web-based interface for defining a component. At the top, there are several tabs: **General**, **Pages**, **PeopleTools** (which is the active tab), **Process**, **Sign-on Times**, and **Component Interfaces**. Below the tabs, the **Permission List** is set to **ALLPAGES**, and the **Description** is **All pages and weblibs**.

The **PeopleTools Permissions** section contains a list of permissions with checkboxes:
 

- ☒ Application Designer Access
- ☒ Data Mover Access
- ☒ Definition Security Access
- ☒ Query Access
- ☐ Performance Monitor PPMI Access

 There are also links for [Definition Permissions](#), [Tools Permissions](#), and [Miscellaneous Permissions](#).

The **Realtime Event Notification** section has a link for [Realtime Event Notification Permissions](#).

The **Data Archival** section contains a 2x2 grid of permissions, all of which are checked:
 

- Generate SQL
- Edit SQL
- Run SQL
- Purge Audit

At the bottom of the page, there are four buttons: **Save**, **Return to Search**, **Add**, and **Update/Display**.

Sample component with multiple pages (tabs)

If a component contains more than one page, you can control which page appears first by setting the display order in PeopleSoft Application Designer. At runtime, the user navigates among the pages of a component by selecting:

- Folder tabs at the top of the page.
- Links at the bottom of the page.
- Specified access or hot keys.

## The Component Buffer

When you open any page in a component, the system retrieves all of the data records for the component and stores them in one set of record buffers organized by scroll level and then by page level. The component buffer consists of rows of buffer fields that hold data for the various records that are associated with page controls, including the primary records, related display records, derived work records, and Translate Table records.

You can change the records in memory either through the user actions on the page or through the PeopleCode that is associated with the different page controls. At runtime, you open and save the entire component, not just the individual pages. When you save any page in a component, you automatically save the whole component.



**See Also**

*PeopleTools 8.52: PeopleCode Developer's Guide*, "Referencing Data in the Component Buffer"

---

## Defining Components

This section provides an overview of the component definition window and discusses how to:

- Create new component definitions.
- Open existing component definitions.
- Add pages to components.
- Reorder pages in a component.
- Copy or move pages to another component.
- Set page attributes.
- Validate components.

## Understanding the Component Definition Window

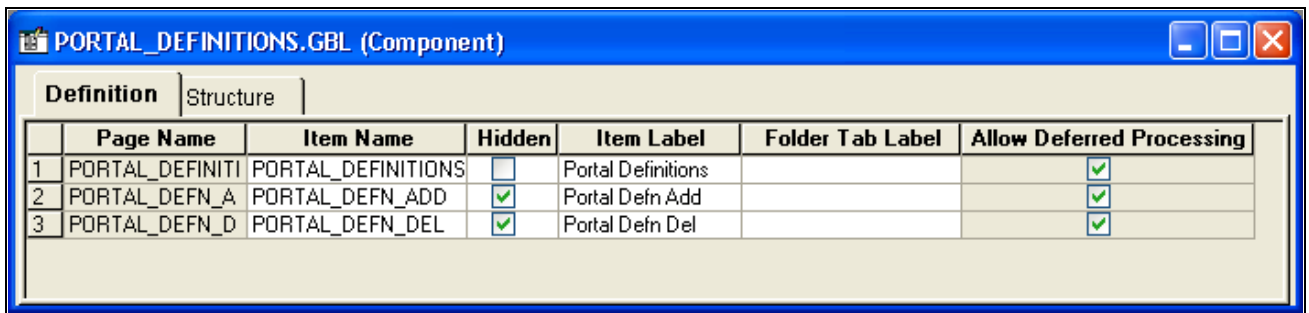
When you design component pages, the pages must share the same Level 0 key structure because they share the same search record.

Depending on the complexity of a component, the system may take longer to display the first page than to display other pages in the component (or even display the first page again) because it is loading records for the entire component.

The component definition window has two tabs, Definition and Structure, that provide different views of a component.

### **Definition Tab**

The default component definition view displays the page items and corresponding attributes of a component definition. You double-click the page name to open the page definition:

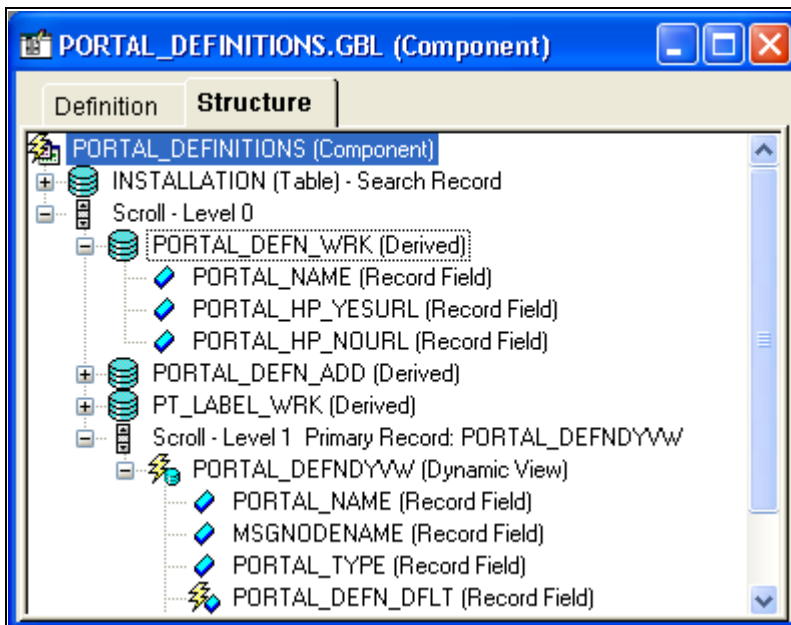


	Page Name	Item Name	Hidden	Item Label	Folder Tab Label	Allow Deferred Processing
1	PORTAL_DEFINITI	PORTAL_DEFINITIONS	<input type="checkbox"/>	Portal Definitions		<input checked="" type="checkbox"/>
2	PORTAL_DEFN_A	PORTAL_DEFN_ADD	<input checked="" type="checkbox"/>	Portal Defn Add		<input checked="" type="checkbox"/>
3	PORTAL_DEFN_D	PORTAL_DEFN_DEL	<input checked="" type="checkbox"/>	Portal Defn Del		<input checked="" type="checkbox"/>

Component definition view

### Structure Tab

The structure view shows records and scrolls in a tree representation. It provides information about page levels and component records and their roles. You double-click the objects in this view to open their definitions:



Component structure view

To view the PeopleCode that is attached to any of the components, right-click and select the View PeopleCode option from the pop-up menu. The PeopleCode Editor opens directly in the workspace, displaying the related PeopleCode.

**Note.** The Structure tab only displays the runtime state of the PeopleCode. That is, it only displays record field PeopleCode. For example, PeopleCode programs that are orphaned as a result of a page definition change do not appear in the Structure tab. On the other hand, these orphaned PeopleCode programs do appear in the PeopleCode Editor, which displays the design-time view of PeopleCode.



A key icon appears next to all key and alternative search key fields in the component structure view.



An asterisk icon appears next to all fields that are required.

## Creating New Component Definitions

To create a new component definition:

1. With a new or existing project open in PeopleSoft Application Designer, select File, New.
2. Double-click Component.

The new component definition appears in the workspace.

When saving a component definition, make sure to adhere to the accepted naming conventions for PeopleSoft definitions.

See [Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33](#).

## Opening Existing Component Definitions

To open an existing component definition:

1. Select File, Open.
2. Select Component as the type.
3. Specify the selection criteria.

The component definition appears in the workspace.

## Adding Pages to Components

This section discusses how to:

- Use the Insert menu to add a page.
- Drag a page into the component.

---

**Note.** The system does not allow you to move or delete all pages from a component definition. The component must contain at least one page definition.

---

### *Using the Insert Menu to Add a Page*

To use the Insert menu to add a page:

1. Open a new or existing component in the workspace and make the definition active.
2. Select Insert, Page into Component.
3. To narrow your search, enter selection criteria, such as name, description, or project.

4. Click Insert.

A list of available pages matching your search criteria appears.

5. Select the page that you want to add to the component.

6. Click Insert.

The page that you selected appears in the component in the workspace.

7. When you are finished adding pages to your component, click Close.

8. Save your component.

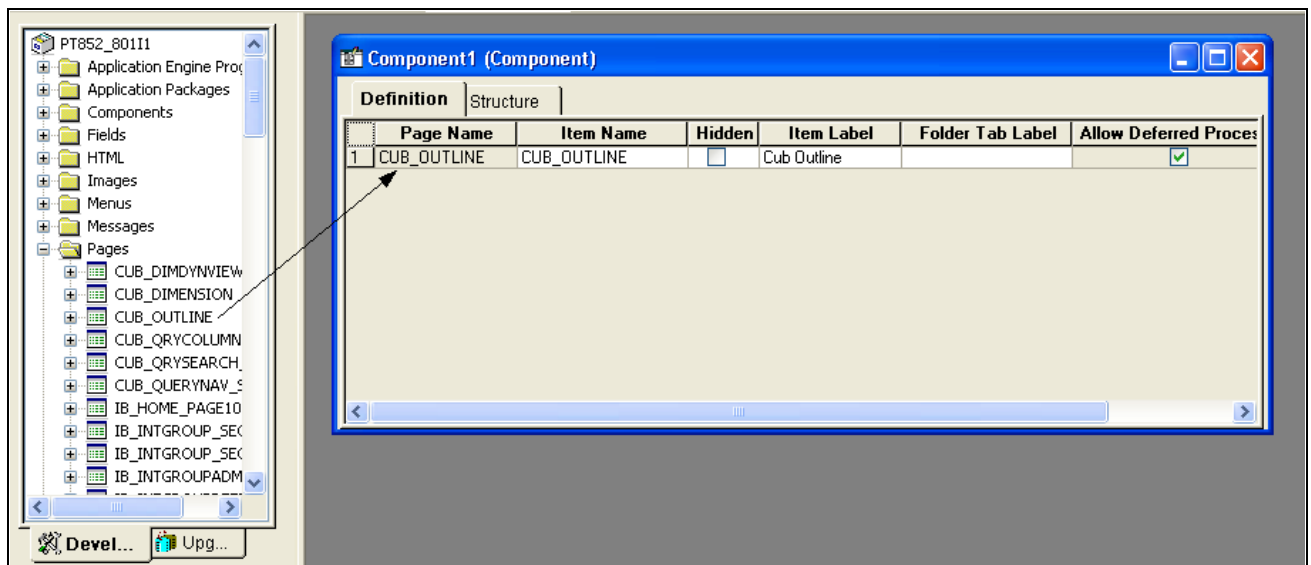
Component names can be up to 18 characters in length.

### ***Dragging a Page Into a Component***

To drag a page into a component:

1. Open the project and component.
2. Drag pages from the project workspace to the component.

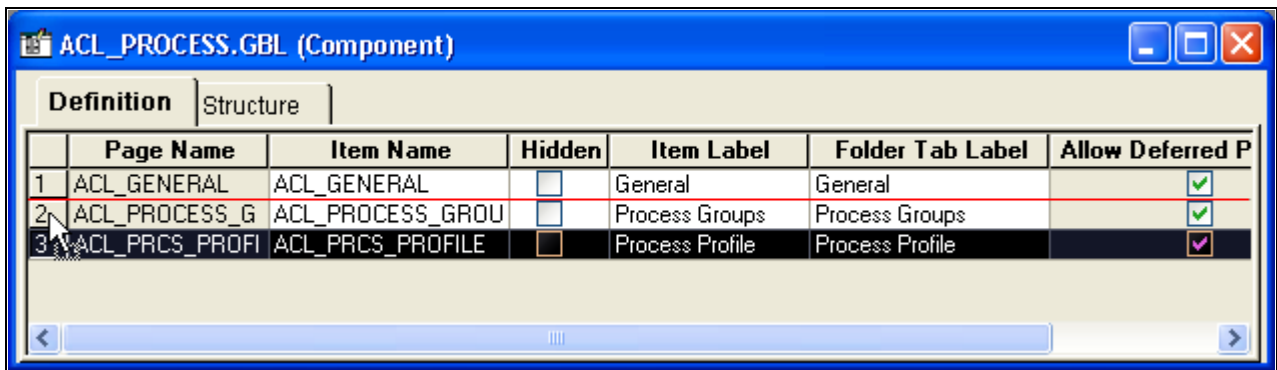
This is an example of dragging a page into a component definition:



Dragging from the project workspace to a component

## **Reordering Pages in a Component**

After you add pages to a component, you can change the order in which they appear in the component.



Moving a page in a component

To reorder a page in a component:

1. Select a row number on the left-hand side of the component definition.
2. Drag the row to the correct position.

The page is inserted immediately below the highlighted line. In the preceding example, the third page (UK) becomes the second page.

## Copying or Moving Pages to Another Component

After you create a component, you can copy or move pages from one component to another.

---

**Note.** The system does not allow you to move or delete all pages from a component definition. The component must contain at least one page definition.

---

To copy or move a page from one component to another:

1. Open both components.
2. In the left-hand column, select the number of the page that you want to copy.
3. Copy or cut the page:
  - To copy the page to another component, select Edit, Copy.
  - To move the page to another component, select Edit, Cut.
4. Select the other component.
5. Select Edit, Paste.

## Setting Page Attributes

Each page in a component has attributes. Each attribute is represented by a column in the component definition:

	Page Name	Item Name	Hidden	Item Label	Folder Tab Label	Allow Deferred Processing
1	AMM_OVERVIEW	AMM_OVERVIEW	<input type="checkbox"/>	Overview		<input checked="" type="checkbox"/>
2	AMM_PUBLIST	AMM_PUBLIST	<input type="checkbox"/>	Message Instances		<input checked="" type="checkbox"/>
3	AMM_PUBCONLIST	AMM_PUBCONLIST	<input type="checkbox"/>	Pub Contracts		<input checked="" type="checkbox"/>
4	AMM_SUBCONLIST	AMM_SUBCONLIST	<input type="checkbox"/>	Sub Contracts		<input checked="" type="checkbox"/>
5	AMM_SYNCMSGLIST	AMM_SYNCMSGLIST	<input type="checkbox"/>	Synchronous Messa		<input checked="" type="checkbox"/>
6	AMM_CHNL_STAT	AMM_CHNL_STATUS	<input type="checkbox"/>	Channel Status		<input checked="" type="checkbox"/>
7	AMM_NODE_STAT	AMM_NODE_STATUS	<input type="checkbox"/>	Node Status		<input checked="" type="checkbox"/>
8	AMM_MULTIDOM	AMM_MULTIDOM	<input type="checkbox"/>	Domain Status		<input type="checkbox"/>
9	AMM_STATISTICS	AMM_STATISTICS	<input type="checkbox"/>	Statistics		<input checked="" type="checkbox"/>
10	PT_APMSGQRY_P	PT_APMSGQRY_PG	<input type="checkbox"/>	Queries		<input type="checkbox"/>
11	AMM_SHARED	AMM_SHARED	<input checked="" type="checkbox"/>	Amm Shared		<input checked="" type="checkbox"/>
12	IB_AMM_FAILOVE	IB_AMM_FAILOVER	<input checked="" type="checkbox"/>	Ib Amm Failover		<input type="checkbox"/>

Component item label and tab label

**Page Name**

Displays the read-only name of the page definition. If you rename the page, the system updates this column automatically.

When creating page definitions, you might want to use similar names to make them easily recognizable as a group of pages. For example, if you create three pages to hold the information for the Personal Data Table, the page names might be:

- PERSONAL\_DATA1
- PERSONAL\_DATA2
- PERSONAL\_DATA3

**Item Name**

Displays a name for each page in the component. This name must be unique in the component, and the default is the page name. This name is for informational purposes only. If you use the same page for more than one component, you might want to change the item name to reflect the purpose of the page in the component.

**Hidden**

Specifies whether the page can be viewed by the user at runtime. Pages are usually hidden when they are used in work groups or associated with derived work records. For example, information from a page might need to be loaded into the buffer for PeopleCode to perform calculations, but the user does not need to see it.

---

**Note.** Marking the page as hidden also hides the toolbar, including the Save button, even if you choose to display the page by means of another method, such as the Transfer PeopleCode function.

---

**Item Label**

Serves as the default folder tab label unless you specify a different label. The item label should be unique for each page in a single component menu. The folder tab label is usually used when shorter names are needed for folder tabs.

In the preceding example (BANK\_ACCOUNT.GBL), the same text appears on both the folder tab and the component menu for the first page in the component. The second and third pages have different menu names and folder tabs.

**Folder Tab Label**

Displays the text that appears on the folder tab label. If no text is specified, the system uses the text of the item label on the folder tabs.

**Allow Deferred Processing**

Indicates whether deferred processing is active for that page in its property settings.

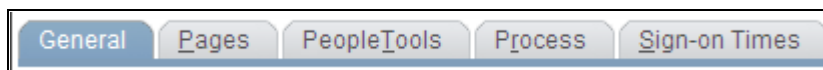
**Setting Access Keys**

You can set access keys by placing an ampersand (&) in the text of each item label. Put the ampersand in front of the letter that you want to be underlined on the folder tab at runtime:

	Page Name	Item Name	Hidden	Item Label	Folder Tab Label
1	ACL_GENERAL	ACL_GENERAL	<input type="checkbox"/>	&General	&General
2	ACL_MENU2	ACL_MENU2	<input type="checkbox"/>	&Pages	&Pages
3	ACL_MISCTOOLS	ACL_MISCTOOLS	<input type="checkbox"/>	People&Tools	&PeopleTools
4	ACL_PROCESS	ACL_PROCESS	<input type="checkbox"/>	P&rocess	P&rocess
5	ACL_SIGNON2	ACL_SIGNON2	<input type="checkbox"/>	&Sign-on Times	&Sign-on Times

Item labels with ampersands

At runtime, a user can navigate between pages by pressing the Alt key plus the letter that is underlined, followed by the Enter key.



Underlined access keys on tabs

**See Also**

*PeopleTools 8.52: PeopleSoft Accessibility Guide, "Using PeopleTools to Build Accessible Applications"*

**Validating Components**

PeopleSoft Application Designer includes a validate utility to verify the integrity of components. The validate utility runs a series of tests on components to check that they will operate properly at runtime, and it sends the results to the Validate tab in the output window.

To validate a component:

1. Open the desired component definition.
2. Select Tools, Options, Validate and make sure that the Validate PeopleSoft Internet Architecture Execution option is selected.

---

**Note.** This option must be selected for the validate tests to be invoked.

---

3. Select Tools, Validate Component.
4. View the results that appear on the Validate tab and correct any errors.

### See Also

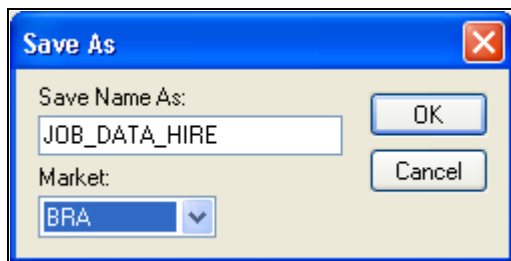
[Chapter 2, "Using PeopleSoft Application Designer," Setting PeopleTools Options, page 22](#)

[Chapter 3, "Working With Projects," Validating Projects, page 60](#)

---

## Specifying Market-Specific Components

You might want to add custom features to a component that is specific to an international market or region. For example, suppose that a page is used in a procedure for hiring new employees and that the procedure includes special transactions that are required only when an employee is hired in Brazil. You can create a custom component using the same component name as the component from which it is derived and saving it using a market setting of *BRA*:



Component saved with BRA Market setting

Market-specific components are independent of system-side language settings and are accessible to any users who have security access to the component. This setting enables an English-speaking user in New York, for example, to perform a procedure for hiring an employee in Brazil.

Access the market setting of a component using the PeopleCode `%Market` system variable. Consequently, you can maintain a single set of PeopleCode programs for a set of market-specific components and build conditional logic into the PeopleCode programs that run only in components with specific market settings.

### See Also

*PeopleTools 8.52: PeopleCode Language Reference*, "System Variables," `%Market`



---

## Locating Component Definition References In Menus

The Find Definition References feature enables you to generate a list of the menus in the database that reference a specific component; that is, a list of the menus in which the component is used. You can invoke this feature from either the project workspace or the definition workspace. To invoke this feature, select Find Definition References from the Edit menu.

### See Also

Chapter 2, "Using PeopleSoft Application Designer," Finding Definitions, page 36

---

## Setting Component Properties

In the Component Properties dialog box, specify the update and data entry actions; on the Use tab, specify the search record information. You can save notes about the component on the General tab. Use the Internet tab to set attributes that affect how the component appears in the browser at runtime.

This section discusses how to:

- Open the Component Properties dialog box.
- Set general properties.
- Set use properties.
- Set internet properties.

## Opening the Component Properties Dialog Box

To open the Component Properties dialog box:

1. Open the component definition.
2. Select File, Definition Properties.

The Component Properties dialog box appears with the General tab active.

## Setting General Properties

Access the Component Properties dialog box.

<b>Description</b>	Enter a descriptive name for the component.
--------------------	---

**Owner ID**

Select the application to which this component belongs. This list is helpful to identify the applications that are associated with the component during the application development phase.

**Last Updated**

View the date and time of the last modification made to the component and the name of the user who made the modification.

## Setting Use Properties

Access the Component Properties dialog box and select the Use tab.

The screenshot shows the 'Component Properties' dialog box with the 'Use' tab selected. The dialog has three tabs: 'General', 'Use', and 'Internet'. The 'Use' tab is active, showing various configuration options. On the left, under 'Access', there are four dropdown menus for 'Search record:', 'Add search record:', 'Detail page:', and 'Context search record:'. Below these is a checkbox for 'Force Search Processing'. On the right, under 'Actions', there are six checkboxes: 'Add' (checked), 'Update/Display' (checked), 'Update/Display All' (unchecked), 'Correction' (unchecked), 'Disable Saving Page' (unchecked), and 'Include in Navigation' (checked). Below the 'Actions' section, there are two sections: '3-Tier Execution Location' and 'Component Save'. Each section contains three radio buttons: 'Client', 'Application server', and 'Default (application server)'. In both sections, 'Default (application server)' is selected. At the bottom right, there are 'OK' and 'Cancel' buttons.

Component Properties dialog box: Use tab

## Access

<b>Search record</b>	Specify the search record for this component.
<b>Add search record</b>	Specify if you want a different search record specifically for add actions. For example, if you selected the auto-numbering option for employee IDs (EMPLID), do not include EMPLID in the search record. Likewise, you might want to create special security views for add actions that limit the rows that users can add based on specific search criteria. The system default is the standard search record if you do not specify an add search record.
<b>Force Search Processing</b>	Select to always run search logic (SearchInit PeopleCode) for this component. The default is deselected.
	<hr/> <p><b>Note.</b> If you select this option, all persistent search functionality for the component is disabled, regardless of system settings.</p> <hr/> <p>See <i>PeopleTools 8.52: PeopleTools Portal Technologies</i>, "Configuring the Portal Environment," Recent Search Results.</p>
<b>Detail page</b>	No longer used in PeopleSoft applications. This option appears for upgrade purposes only.
<b>Context search record</b>	Specify the record that defines the service parameters that you must map for PeopleSoft component URL-related content services or content reference service types. The system uses the context search record only when no key fields are defined for the primary search record and only in a related content configuration.
	See <i>PeopleTools 8.52: PeopleTools Portal Technologies</i> , "Developing and Configuring Related Content Services," Configuring Service Parameters.

---

**Note.** To see a subset of search records, type a few characters first. The drop-down list box is populated with a list of all record definitions with names that begin with those characters, thus reducing access time and enabling you to find specific records faster.

---

## Actions

<b>Add</b>	Select to add a new high-level key, such as a new employee ID or customer. Except in the case of effective dating, use Add to insert a new current row or to update future rows.
<b>Update/Display</b>	Select to update existing rows only.
<b>Update/Display All</b>	Select to update current and future rows in an effective-dated record. Use only with effective-dated records. Do not use these actions unless the main record that is associated with the page definitions is effective dated. This information is translated to include history at runtime.

<b>Correction</b>	Select to update any rows (history, current, and future) in an effective-dated record. Use only with effective-dated records. This information is translated to correct history at runtime.
<b>Disable Saving Page</b>	Select when you want to hide the Save button in the toolbar and disable the Alt+1 (Save) hot key. Selecting this option prevents users from being prompted to save when exiting a page. However, it does not prevent using PeopleCode to save a page with the <code>DoSave ( )</code> or <code>DoSaveNow ( )</code> functions. This functionality can be helpful for applications in which the user is not making database changes and does not need to be prompted to save.
<b>Include in Navigation</b>	Select to include the component in the menu navigation at runtime. The default is selected. If you do not want the component included, deselect the check box.
<b>Mandatory Spell Check</b>	Select to ensure that spell check is run on all eligible spell check enabled fields in the component when a user saves the page. A field is eligible if it has been changed (or its ReEdit flag has been set) and it has not already been manually spell checked since it was last changed.

### Read Only

---

**Note.** If you must add a new high-level key, such as a new employee number or customer, select both Add and Update/Display. If you are adding information to an existing high-level key, such as adding education data for an employee, select only Update/Display. You are not adding a new employee ID; it already exists on the database.

---

### *Three-Tier Execution Location*

This section applies only to Windows client applications.

To partition application processing between the client and the application server, you must define units that, as a whole, run in one location or the other. We call these units processing groups. Processing groups can encompass one or more PeopleCode events.

<b>Component Build</b>	Specify where you want all processing to occur after the key list of a page is selected and before the user can interact with the page. Processing includes building component buffers and running many types of PeopleCode.
<b>Component Save</b>	Specify where you want all processing to occur after the user saves the component and SaveEdit PeopleCode validations have succeeded. Processing includes SavePreChange, WorkFlow, SavePostChange PeopleCode, and updates to the database.

## Setting Internet Properties

Access the Component Properties dialog box and select the Internet tab.

**Component Properties**

General Use **Internet**

**Search Page**

Primary Action: ☐ New ☒ Search ☐ Keyword Search

Default Search Action:

Default Search/Lookup Type: ☒ Basic ☐ Advanced ☐ Keyword Only

☒ Allow Action Mode Selection

Link To Add Page	Message Set/Nbr:	<input type="text" value="124"/>	<input type="text" value="62"/>
Link To Find Existing Value Page	Message Set/Nbr:	<input type="text" value="124"/>	<input type="text" value="63"/>
Link To Realtime Search Page	Message Set/Nbr:	<input type="text" value="124"/>	<input type="text" value="63"/>
Link To Keyword Search Page	Message Set/Nbr:	<input type="text" value="124"/>	<input type="text" value="433"/>
Instructional Text	Message Set/Nbr:	<input type="text" value="124"/>	<input type="text" value="50"/>

**Multi-Page Navigation**

☒ Display Folder Tabs (top)

☒ Display Hyperlinks (bottom)

**Processing Mode**

☒ Interactive ☐ Deferred

☐ Allow Expert Entry

☐ WSRP Compliant

**Toolbar**

Selected Toolbar Actions:

<input checked="" type="checkbox"/> Save	<input type="checkbox"/> Refresh
<input checked="" type="checkbox"/> Cancel	<input checked="" type="checkbox"/> Notify
<input type="checkbox"/> Spell Check	<input checked="" type="checkbox"/> View WorkList
<input checked="" type="checkbox"/> Return to List	<input checked="" type="checkbox"/> Next in WorkList
<input checked="" type="checkbox"/> Next in List	<input checked="" type="checkbox"/> Previous in WorkList
<input checked="" type="checkbox"/> Previous in List	<input checked="" type="checkbox"/> Add
<input type="checkbox"/> Next Page in Component	<input checked="" type="checkbox"/> Update/Display
<input type="checkbox"/> Previous Page in Component	<input type="checkbox"/> Update/Display All
<input type="checkbox"/> Disable Toolbar	<input type="checkbox"/> Correction

**Pagebar**

☒ Help Link

☒ Copy URL Link

☒ New Window Link

☒ Customize Page Link

☐ Disable Pagebar

OK Cancel

Component Properties dialog box: Internet tab

## Search Page

### Primary Action

Select the default mode by which the user accesses the component.

Select New to access the component in Add mode.

Select Search to access the component in the mode that is specified in the Default Search Action field.

Select Keyword Search to access the component using the Keyword Search page. The Keyword Search page is available only if you have configured the PeopleSoft Search Framework and mapped the component to a search definition.

See *PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Working with Search Pages.

Specify where you want all processing to occur after the key of a page is selected and before the user can interact with the page. Processing includes building component buffers and running many types of PeopleCode.

### Default Search Action

Select from a list of the actions that you set in the Actions group box on the Use tab of the Components Properties dialog box. If you set only one action, this drop-down list box provides that choice as the default. The default search action determines the default that users see, but they can select another option from the toolbar at runtime.

### Default Search/Lookup Type

Specify the type of search or lookup which appears by default when users access the search page and lookup pages associated with the component.

Select Basic display the Basic search page by default. The Basic search page provides access to a list of search key and alternate search key fields from which the user can choose one field to search. You should select Basic as the default search type for new users because a basic search page is easier to use than other types of search pages.

Select Advanced to display the Advanced search page by default. The Advanced search page provide access to all search key and alternate search key fields simultaneously. You should select Advanced as the default search type for power users because an advanced search page enables complex searches.

Select Keyword Only to display just the Keyword Search page by default. The Keyword Search page is available only if you have configured the PeopleSoft Search Framework and mapped the component to a search definition.

See *PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Working with Search Pages.

<b>Allow Action Mode Selection</b>	Select to display the Include History and Correct History check boxes on a search page. If a user selects Include History on a search page, the page appears in Update/Display All mode. If a user selects Correct History, the page appears in Correction mode. However, both of these check boxes appear only on the search page if Update/Display All and Correction were selected as actions in the Use properties for the component and the user has security access to the modes. This check box is selected by default.
<b>Link to Add Page</b>	Enter the message set and number that stores the text that comprises the link which users click to access the standard add page.
<b>Link to Find Existing Value Page</b>	Enter the message set and number that stores the text that comprises the link which users click to access the standard search page.
<b>Link to Realtime Search Page</b>	Enter the message set and number that stores the text that comprises the link which users click to access the Realtime Search page. Because Realtime search functionality is disabled by default, these values are the same as the values associated with the Link to Find Existing Value Page. However, if you enable Keyword Search, you can enter different values so that the link text accurately indicates the type of search page that users access via the link.
<b>Link to Keyword Search Page</b>	Enter the message set and number that stores the text that comprises the link which users click to access the Keyword Search page, which appears only when Keyword Search is enabled for the component.
<b>Instructional Text</b>	You set this option to display the Find an Existing Value heading on the search page. You can change the default message or redirect the component to use another message that is more appropriate for your use.

## ***Toolbar***

Select the toolbar actions that you want to appear on the component pages. When you select a check box, that item appears as a toolbar button at the base of the component at runtime. All toolbar actions are set to display by default with the exception of Next Page in Component, Previous Page in Component, and Notify. Note that the default settings for the Action and Save buttons are affected by settings on the Use tab of the Component Properties dialog box.

Disabling any of the toolbar options also disables the hot key for that option in the PeopleSoft Pure Internet Architecture.

<b>Save</b>	Adds a Save, OK, or Apply button to the toolbar, depending on the page type.
<b>Cancel</b>	For modal components only. Deselecting this check box removes the Cancel button from the toolbar.
<b>Spell Check</b>	Runs spell check for all spell check-enabled fields in the component, regardless of whether any changes were made to the field.
<b>Return to List</b>	Returns users to the search page.

<b>Next in List and Previous in List</b>	Populates the page with the next or previous row of data from the database as shown on the Search page.
<b>Next Page in Component and Previous Page in Component</b>	Opens the next or previous page in the component if it has more than one page.
<b>Refresh</b>	Displays the page again without saving any entered data.
<b>Notify</b>	Enables the user to send a notification to other users.
<b>View Worklist</b>	Enables the user to view the worklist from the current component page.
<b>Next in Worklist and Previous in Worklist</b>	Enables the user to access the next or previous item in the worklist.
<b>Add, Update/Display, Update/Display All, and Correction</b>	Select each option to enable these toolbar buttons for this component. See <a href="#">Chapter 12, "Creating Component Definitions," Setting Use Properties, page 404.</a>
<b>Disable Toolbar</b>	Select if you do not want the toolbar to appear on the component pages.

### ***Multi-Page Navigation***

<b>Display Folder Tabs</b>	Select to display folder tabs at the top of the component page.
<b>Display Hyperlinks</b>	Select to display links at the bottom of the component page.

### ***Processing Mode***

<b>Interactive</b>	Select to send messages to the server whenever logic must be run. When the user presses Tab to exit a field-level event, a transmission to the application server occurs to run that field-level event and the page is refreshed.
<b>Deferred</b>	Selected as the default. This mode reduces transmissions to the application server.  See <a href="#">Chapter 8, "Designing Pages for Optimal Performance," Understanding Processing Modes, page 199.</a>
<b>Allow Expert Entry</b>	This option is available for both processing modes.  See <a href="#">Chapter 12, "Creating Component Definitions," Enabling the Expert Entry Feature in the Security Component, page 420.</a>



### **WSRP Compliant Option**

Select the WSRP Compliant option to enable the component for Web Services for Remote Portlets (WSRP) publishing. If the component is enabled for WSRP publishing, then users can render the component in-line within the WSRP consuming portal.

If the component is *not* enabled for WSRP, when users navigates to the component from within a WSRP portlet the system will direct them to the PeopleSoft Portal, which loads the component. The user will effectively leave the WSRP consuming portal page and navigate to the PeopleSoft system.

---

**Note.** This option enables you to distinguish between components that are known to function only in the PeopleSoft Portal and those that are WSRP-compliant.

---

See *PeopleTools 8.52: PeopleTools Portal Technologies*, "Understanding Web Services for Remote Portlets (WSRP)."

### **Pagebar**

The pagebar is a small menu bar that appears at the top of each component page just below the universal navigation header. You can select which options appear to the user for each component. All options are selected by default. Select the Disable Pagebar check box to disable the pagebar.

#### **Help Link**

Opens the online help PeopleBook entry for the current page.

#### **Copy URL Link**

Copies the current page URL to the clipboard. The copied URL includes page, action, and search key information that appears in the context of the portal if the user passed through the portal. If the user did not pass through the portal, then only the page appears, without the portal frames.

This option adds a small http button to the pagebar.

#### **New Window Link**

Opens a new browser window with the search page for the current component. Users can view or enter data in the new window.

**Personalize Page Link**

Opens the Page Personalizations page, which allows users to control the initial appearance of the component. This option enables users to:

- Select the initial page in the component that appears.
- Save the state of the expanded and collapsed sections.
- Save the state of the View All settings on the page for grids and scroll areas.
- Personalize the page field tab order.

Users can also elect to copy and share their settings for the component with other users.

---

**Note.** If you set this option for a particular component, all users accessing the component are able to personalize the pages in the component unless their personalization options prevent them from customizing pages. Security permission list settings or personalization options do not allow the option to be enabled or disabled per end user per component.

---

See *PeopleTools 8.52: Security Administration*, "Managing PeopleSoft Personalizations," Understanding General Options.

See *PeopleTools 8.52: PeopleSoft Applications User's Guide*, "Setting User Preferences," Personalizing Pages.

You can also enable or disable any of these options using PeopleCode.

See *PeopleTools 8.52: PeopleCode API Reference*, "Page Class."

**Personalize Page Option**

For the page personalization option to appear to users, the following criteria, in addition to the selection of the Page Customization Link in the Pagebar, must be met:

- A PeopleSoft Pure Internet Architecture page, including search and secondary pages, must be visible to the user.
- The page cannot be a PeopleTools-generated page, such as the Page Personalization page or the Personalize Column and Sort Order page for grids.
- The current page cannot be an iScript.
- The Personalize Pagelink cannot be hidden using PeopleCode for the current component.
- The Personalize Page Settings, CUSTOMPGSET, personalization option on the Define Personalizations page must be enabled.

This option is enabled by default. If this option is disabled, the end user cannot define, share, or copy page personalizations and all existing page personalizations for the user are deleted.

- The current page must allow some form of personalization based on the end user's security profile and meet *at least one* of these conditions:
  - Display more than one page element with a TABINDEX  $\geq 1$  supporting personalization (assume that all collapsible sections are expanded when determining visibility).
  - Contain at least one visible scroll area or grid with a visible View All link, enabled or disabled.
  - Contain at least one visible collapsible section.

### See Also

*PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Understanding Search Pages

*PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Enabling Keyword Search

*PeopleTools 8.52: PeopleSoft Search Technology*, "Working with PeopleSoft Search," Developing for Search Pages

---

## Using Search Records

This section provides overviews of search records and search records and persistent search results, and discusses how to:

- Create search records.
- Override search records.

## Understanding Search Records

You must correctly configure and select the search record for a component because the system uses the search record for several purposes:

- The search record builds the search page for the component.
- The search record builds the search query, which retrieves the component data and populates level 0 for every page in the component.
- The search record builds the recent search results page for persistent searches.
- The search record defines the service parameters that must be mapped for related content service definition of the PeopleSoft Component URL type.

When you click a content reference to access a component, the system examines the search record and dynamically builds the search page based on the search key, alternate search key, and list box item properties of the search record. If you designate fields as search keys or alternate search keys, then those fields appear on the search page. When the user enters values in the search fields and clicks the Search button, the system incorporates those values into the WHERE clause of the search query. If the values provided by the user uniquely identify one row of data, then the system proceeds directly to the main page with the available information. Otherwise, the query retrieves the rows that match the criteria and displays the rows in a list box at the bottom of the search page. If you define fields as list box items on the search record, the system displays those fields as columns grid in the result set.

This example shows elements of a search page:

The screenshot shows a web application interface for searching 'Resolution' records. On the left is a 'Menu' sidebar with a search field and a tree view containing items like 'My Favorites', 'ICE', 'Search - Configurable', 'Work Resolution', 'Bundles', 'Metrics & Reporting', 'User Setup', 'ICE Build Tracking', 'Test Cases', 'Translation Reviewer', 'ICE Administration', 'Worklist', 'Reporting Tools', and 'PeopleTools'. The main area is titled 'Resolution' and contains a search form with fields for Resolution ID, State, Resolution Type, Bundle ID, Create Date, Created By Name, Resolved By Name, and Subject. Each field has a dropdown menu and a text input. A red box highlights these fields with a note: 'These are Search Key and Alternate Search Key fields on the Search record.' Below the form are buttons for 'Search', 'Clear', 'Basic Search', and 'Save Search Criteria'. The 'Search Results' section shows a message: 'Only the first 300 results can be displayed. Enter more information above and search again to reduce the number of search results.' Below this is a 'View All' link and a table of results.

Resolution ID	State	Resolution Type	Bundle ID	Lead Product	Create Date	Created By ID	Created By Name
870196	Posted	Applic	0	HRMS Global Payroll Core 9.1	08/11/2011	SRINSHAR	Srinivasa Sharma
870085	Posted	Applic	0	FIN Expenses 9.1	08/08/2011	22383	Rafael Almeida
870077	Posted	Applic	0	CRM Enterprise Components 9.1	08/08/2011	15463	Shelly Bowman
870063	Posted	Applic	0	PRTL Applications Portal 8.9	08/08/2011	T12987	Madanagopal Damodharan
870003	Posted	Applic	0	CRM Advanced Configurator 9.1	08/05/2011	C8VPATEL	Vishnukumar Patel
869878	Posted	Applic	0	CRM Online Marketing 9.1	08/02/2011	AAZEEM	Abdul Azeem

These are List Box fields on the Search record. Note that "Lead Product" and "Created By ID" aren't key fields, just Listbox items, so they aren't shown above, just here.

Search page with search keys, alternate search keys, and list box items

As soon as the system retrieves the search results, they appear as rows in the list box or grid. If recent search results caching is enabled, the system also temporarily stores this most recent (or persistent) search in the menu structure—in the navigation path and in the Recent Search Results menu under the Favorites drop-down menu.

When you select a row from the list box, the runtime system generates a second query to fill the level 0 row in the component. The runtime system examines the level 0 record for the component and checks its key fields. It tries to retrieve level 0 key field values from the fields in the list box and uses those values in the WHERE clause used to populate the level 0 record.

---

**Note.** This process is the same whether you select the row from the initial search results list or from a persistent search.

---

The search record for a component might differ from the level 0 record definition for a component and might contain any number of search keys and list box items, but it must contain all of the key fields that are in the level 0 record for a component.

See [Chapter 6, "Creating Record Definitions," Setting Record Field Use Properties, page 134.](#)

---

**Note.** When you add page items that are based on derived work records, do not select that record definition as the search record. Instead, select a record definition that stores the primary key or a SQL view that searches for the criteria that best enables users to identify the rows that they want to retrieve.

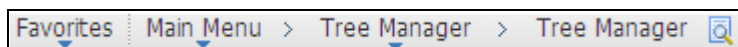
The search record might also contain logic to refine the search and limit rows of data. PeopleSoft applications implement data permission (or row-level) security in this way.

To configure menus in a PeopleSoft application, the easiest way to decide which search record to use is to look at the other search records that are used on the same menu. Look for pages with similar key structures and then evaluate the search records to see if any are suitable. You might want to print the PeopleTools Menu Listing cross-reference report (XRFMENU) to review which search records are used for pages and gain a better understanding of how search records are assigned in the application.

---

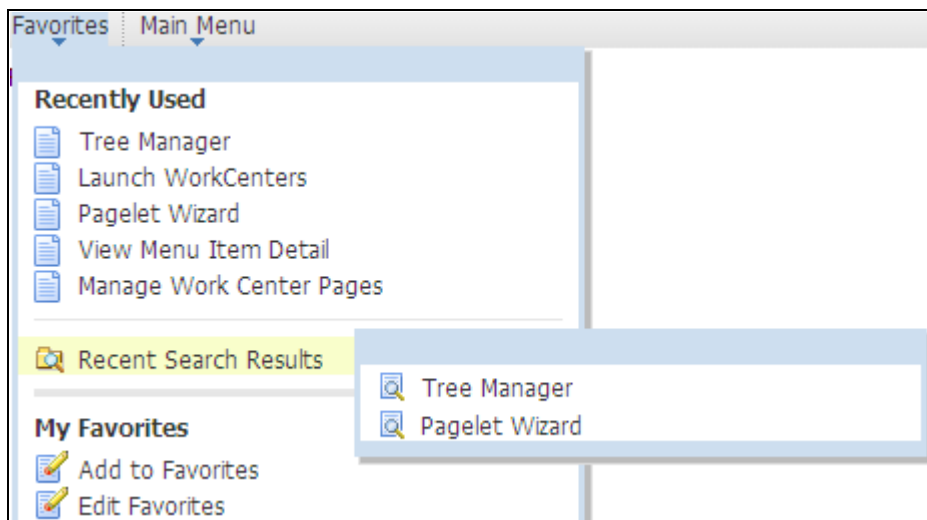
## Understanding Search Records and Recent (Persistent) Search Results

When you perform a component search, the results of your most recent search are temporarily stored—they persist—in the portal drop-down menu structure. The system stores the most recent component search results for the currently active component in the navigation path (breadcrumbs). Here is an example of the search results icon in the navigation path:



Search icon at the far right in the navigation path

The most recent search for each of the five most recently visited components appear in the Recent Search Results menu:



Recent Search Results menu showing two recent component searches

---

**Note.** Components that do not generate search pages do not generate search results and, therefore, do not appear in the Recent Search Results menu or the navigation path.

---

When you access a recent search either from the navigation path or the recent search menu, the system displays the results in a grid on a modal pop-up page, which hovers over the currently active page.

Here you see the Recent Search Results pop-up page for the Tree Manager component:

Recent Search Results

Tree Manager

[Search Again](#)

View All | First ◀ 1-20 of 20 ▶ Last

Tree Name	SetID	Set Control Value	Effective Date	Tree Branch
<a href="#">QE ACCOUNTS</a>	QEDM2	(blank)	1900-01-01	(blank)
<a href="#">QE ACE 848 EMPL</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE AFTREE</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC BU</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC DEPT</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC DTRG1</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC EMPD2</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC EMPD3</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC EMPD4</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC PRDID</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE DGC PROD</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE PRODUCT</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE ACE PROD TREE</a>	QEDM1	(blank)	1900-01-01	(blank)
<a href="#">QE CUBT PRODUCTS</a>	QEEU	(blank)	2009-02-27	(blank)
<a href="#">QE CUBT REGION</a>	(blank)	(blank)	2009-03-01	(blank)
<a href="#">QE DEPT DYNAMIC</a>	QEDM1	(blank)	1999-01-01	(blank)
<a href="#">QE DEPT SEC</a>	QEDM2	(blank)	2001-01-01	(blank)
<a href="#">QE JOBCODES</a>	(blank)	(blank)	1999-01-01	(blank)
<a href="#">QE PERS DATA</a>	QEDM1	(blank)	1997-05-05	(blank)
<a href="#">QE PROJECTS</a>	(blank)	QEBU1	2000-04-25	(blank)

Tree Manager Recent Search Results pop-up page showing results grid

---

**Important!** You must enable recent search results caching at the system level. The default setting for recent search results caching is disabled.

---

### ***Recent Search Results Process***

A persistent search enables you to circumvent the first part of the component search process by temporarily storing all search keys, which the system uses in the WHERE clause of the second part of the search process, in the menu structure. When you access a recent search and select an item from the list, the second query is generated and populates the component buffer.

Although the search process remains the same, the appearance of the search results page differs from component search pages.

Note these points about the search results page:

- It is modal; it hovers over the transaction page, which is inactive until you dismiss the modal page or select a list item.
- It is compact; it displays a maximum of five of the temporarily stored fields in the results grid.

- It is moveable, which enables you to view the current transaction and see the search results simultaneously.
- It displays an active link, a link that you can click to retrieve a specific row of data, in only the first column; all other columns are inactive.
- It is configurable.

See [Chapter 6, "Creating Record Definitions," Configuring Search Results Grid Columns in Persistent Searches, page 141.](#)

Note these points about the accessibility and availability of persistent search results:

- The most recent transaction search results for the currently active component appear in the navigation path next to the content reference name; a search icon is the visual indicator.
- The one most recent search for each of the five most recently viewed components appears in the Favorites, Recent Search Results menu. The results appear by component name, for example, User Profiles.
- The navigation path retains the most recent transaction search results until another search is performed.

---

**Note.** If you leave a component and return to it without having performed another search, the most recent search results reappear.

---

- Any persistent search remains in the portal menu structures only until your PeopleSoft session expires, you sign out, or you close the browser.

### ***Recent Search Results Page Construction***

The standard, default Recent Search Results page uses these criteria to determine the fields that appear on the page:

- Fields marked as search key fields appear in the results grid in the order in which they are listed in the record definition.
- If the search record contains more than five search key fields, the other search key fields, while stored in the menu structure and accessible by the search process, are hidden from view.
- If the search record contains fewer than five search keys, the fields marked as list box items appear in the grid in the order that they are listed in the record definition until the column limit of five is reached.

You can override which columns appear on the recent results search page by using the Persist in Menu property of the search record.

### ***See Also***

[Chapter 6, "Creating Record Definitions," Persist in Menu Configuration, page 140](#)

## **Creating Search Records**

To create a search record:



1. Create the record definition and add the fields.

See [Chapter 6, "Creating Record Definitions," Creating New Record Definitions, page 124.](#)

2. Select the key fields, search key fields, and list box items.

See [Chapter 6, "Creating Record Definitions," Setting Record Field Use Properties, page 134.](#)

3. If the record is a view, write the SQL statement in the SQL editor.

See [Chapter 6, "Creating Record Definitions," Creating SQL View and Dynamic View Select Statements, page 152.](#)

4. Save and build the record definition.

See [Chapter 7, "Administering Data," Running the Build Process, page 156.](#)

## Overriding Search Records

You might want to reuse a component, changing only its search record.

You can accomplish this by overriding the component search record at the time that the component is invoked from a menu. To override a component search record, specify a different search record in the properties of the menu item that invokes the component. The component uses the override search record when it is invoked from that specific menu item; the search record set in the component properties remains unchanged.

By reusing components in this way, you can limit redundancy, keeping your application smaller and easier to maintain.

### See Also

*PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," `SetSearchDialogBehavior`

---

## Accessing the Message Catalog

Message sets and number settings come from the PeopleTools Message Catalog.

To access the Message Catalog:

1. Select PeopleTools, Utilities, Administration, Message Catalog.
2. On the search page, enter the message set number that you want to access.

Component definitions support messages between 0 and 99,999.

3. Click Search to access a Message Catalog page.
4. Enter the message set and number on the Internet tab of the Component Properties dialog box.

For example, enter message set and number *124* and *50* to display the Find an Existing Value message on the search page in the browser.

---

## Enabling the Expert Entry Feature in the Security Component

This section provides an overview of the Expert Entry feature and discusses how to enable it in Security.

### Understanding the Expert Entry Feature

The Expert Entry feature enables users to change from interactive to deferred mode at runtime for appropriate transactions.

---

**Note.** Only expert users who understand how all of the fields in a transaction behave should use the Expert Entry feature. These users must know the exact keystroke sequence for the transaction. This sequence typically varies for a transaction by customer, depending on the customer's setup and entry requirements. Therefore, occasional users should not use the Expert Entry feature because they might not be aware of the nuances of the transaction and when it is necessary to press the Refresh hot key.

---

To use Expert Entry, select:

- The Allow Expert Entry check box on the Internet tab of the Component Properties dialog box.
- The Enable Expert Entry check box in the user's profile.

If these check boxes are selected, an Expert Entry check box appears at the bottom of the page at runtime. If users select this check box, the transaction runs in deferred mode, regardless of the setting on the component, page or field property.

When you set the component property to deferred mode, that component operates in deferred mode processing for all users, not just those who have Expert Entry enabled in their profiles.

If you want the component to process in deferred mode for some users and in interactive mode for other users:

- Set the component property to Interactive mode.
- Select the Allow Expert Entry check box.

These settings enable security user profiles to control the users for whom deferred mode processing is possible.

---

**Warning!** Users who modify component properties from interactive mode to deferred mode or select the Allow Expert Entry check box should understand that this is an adaptation of the software and, as such, they are responsible for thoroughly testing and supporting this change.

---

#### **Refresh Button**

The Expert Entry feature enables you to specify whether a Refresh button should be included on the component toolbar when the system displays the page. The Refresh button forces a transmission to the server to determine which fields have changed since the last transmission to the server and to run any processing logic that is associated with those changes.

Users can also refresh by pressing the Alt+0 hot key, which keeps the cursor in the same field when the page is refreshed. One can refresh any time during data entry to enable an expert users to:

- Update related display field values for data that was already entered.
- Recalculate totals and balances.
- Provide defaults based on data that was entered previously on the page.
- Validate data entered on the page thus far.
- Invoke the code to hide or gray items associated with specific fields.

### ***Limitations of Expert Entry***

These limitations exist when Expert Entry is enabled:

<b>Hide, Unhide, Gray, and Ungrey</b>	The user must click the Refresh button after entering a value in a field that is associated with PeopleCode that hides, unhides, grays, or ungrays other fields on the same page.
<b>Drop-down list box</b>	The user must click the Refresh button after entering a value in a field that is a high-order key that is used to control the values in a drop-down list box.
<b>Default values</b>	Whenever Expert Entry is properly enabled and selected by a user at runtime, the transaction runs in deferred mode. As such, defaults, totals, balances, and related displays are not updated until the next transmission to the application sever (for example, when the user clicks a button, icon, link, or another tab). If a user wants the page updated before then, she or he must click the Refresh button.

## **Enabling Expert Entry in Security**

To have the Expert Entry check box appear in the component at runtime, the security administrator must enable it in Security. The Expert Entry check box will then appear in any components for which you select the Allow Expert Entry property on the Internet tab of the Component Properties dialog box.

To enable Expert Entry in Security:

1. Sign in to the PeopleSoft system in your browser.
2. Select PeopleTools, Security, Use, User Profiles.
3. Select the General tab.
4. Select Enable Expert Entry in the General Attributes group box.
5. Click Save.



## Chapter 13

# Creating Menu Definitions

This chapter provides an overview of menus and discusses how to:

- Define custom component menus.
- Work with menu definitions.
- Define pop-up menus.

---

## Understanding Menus

A menu is a logical grouping of components for assigning security to your system. You create a menu as an organizer for components. After you save a menu, use the Registration Wizard to assign components and security.

You can create two types of menus using PeopleSoft Application Designer:

- Custom (component) menus provide an internal reference for components and pages. They are grouped logically into menu groups to assign security. They are not used for navigational purposes.
- Pop-up menus appear when a user clicks a pop-up button.

### **See Also**

[Chapter 14, "Using the Registration Wizard," page 435](#)

*PeopleTools 8.52: Security Administration, "Understanding PeopleSoft Security"*

---

## Defining Custom Component Menus

This section discusses how to:

- Create custom component menus.
- Set menu item properties.
- Set general properties.
- (Optional) Set use properties.

## Creating Custom Component Menus

After you create records, pages, and components, register your new component using the Registration Wizard. To do so you must first create a menu definition if one does not already exist. This definition enables you to add the component to the new menu (for navigational purposes) and register it in the portal.

To create a new menu definition:

1. From the PeopleSoft Application Designer toolbar, select File, New..
2. Select Menu from the list.
3. Click OK.
4. Select the Standard option for the menu type.
5. Click OK.

The thick rectangle surrounding the bar item indicates that this element in the menu definition is currently selected. The empty, dashed rectangle is a placeholder for a new bar item label.

6. Assign a bar name and label.

You can define multiple menu bars for the menu. Double-click the rectangle to display the Bar Item Properties dialog box.

7. Enter the bar item name and label.

The most commonly used bar item name is Use..

8. Click OK to accept the settings in the Bar Item Properties dialog box.

The new label appears in the menu, and the empty rectangle appears in a new location.

9. Set the menu item properties.

When creating a menu simply as a placeholder for components for the Registration Wizard, you need only add a separator bar to save the menu.

10. Set the menu properties.

11. Save the menu definition.

### See Also

[Chapter 13, "Creating Menu Definitions," Setting Menu Item Properties, page 425](#)

[Chapter 13, "Creating Menu Definitions," Setting General Properties, page 426](#)

[Chapter 14, "Using the Registration Wizard," page 435](#)

[Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33](#)

## Setting Menu Item Properties

Set the menu item properties to define your menu item. When creating a new menu definition to be used in the Registration Wizard, select Separator from the Type area of the Menu Item Properties dialog box to save the menu definition for use in the Registration Wizard. If you decide not to use the Registration Wizard, complete the following procedure, and set the general and use properties for the menu definition.

To define a menu item:

1. In a standard menu definition, double-click a menu item to access its properties.

To create a new menu item, double-click the empty rectangle at the bottom of the menu.

2. Specify the menu item.

<b>Name</b>	Specify the system name for the menu or menu item. Menu and menu item names must conform to a specific set of naming conventions for the system to recognize them: all uppercase letters, no embedded spaces, and no special characters (for example, %, ^, &, *, \$, #).
-------------	---

<b>Label</b>	Specify the text label to appear on the menu. The label appears in the Home bar item or in a cascade menu of a menu group. You can place an ampersand in the text to assign a keyboard shortcut for the menu.
--------------	---

3. Select the menu item type.

<b>Component</b>	Select for a standard component menu item.
------------------	--

<b>PeopleCode</b>	PeopleCode menu items trigger PeopleCode programs. The program runs in a component buffer context, which allows PeopleCode to access values in the component buffer.
-------------------	--

<b>Separator</b>	Select this option when creating a menu for the Registration Wizard. Selecting this option disables all other fields in the Menu Item Properties dialog box.
------------------	--

4. Click the Select button.
5. Select an enabling component in the Open dialog box.

At runtime, standard PeopleCode menu items are always visible in a menu, but they are active (enabled) only when their enabling components are open.

6. Save the menu item if it has not yet been saved.
7. In the menu definition, right-click the menu item, and then select *View PeopleCode*.

The PeopleCode Editor appears.

8. Add a PeopleCode program in the ItemSelected event of the menu item.
9. After you enter a program, save the PeopleCode program and close the PeopleCode editor.

#### 10. (Optional) Override the component search record.

You might want to reuse the same component multiple times with different search records. You can accomplish this by overriding the component search record at runtime when the component is opened from a menu item without creating separate copies of the component. The component override is temporary and occurs only when the component is opened from the menu item in which the override is set. It does not change the component definition.

Select the Override check box, and then select an override search record from the Override prompt field. The override search record must be a valid search record for the component; that is, it must have all of the search key fields that are at level 0 of the component pages.

#### 11. Click OK to accept the settings.

#### 12. Save the menu definition.

## Setting General Properties

When you create or make changes to a menu, you should document the changes for future reference. You can do this on the General tab of the Menu Properties dialog box.

<b>Description</b>	Enter a descriptive name for the menu.
<b>Comments</b>	Add a description of the menu or any other useful information.
<b>Owner ID</b>	View a list of applications with which this menu is used. This list is helpful to identify the applications with which the menu is associated during application development.
<b>Last Updated</b>	View the date and time of the last modification and the name of the user who made the modification.

## (Optional) Setting Use Properties

Setting properties on the Use tab of Menu Properties is optional. Any settings you assign are stored internally only.

---

## Working With Menu Definitions

You should not modify PeopleTools or other application-specific menus; instead, create your own custom menus.

This section discusses how to:

- Add components to a menu.
- Rename menu definitions.
- Copy menu definitions.



- Delete menu items.
- Print menu definitions.
- Set up menu security.
- Import menu groups into portals.

## Adding Components to a Menu

Use the Registration Wizard to add components to a menu.

See [Chapter 14, "Using the Registration Wizard," page 435](#).

## Renaming Menu Definitions

When you change the name of a menu definition, you do not change the name of the application window that appears when you select the new menu from the PeopleSoft application.

To rename a menu definition:

1. Close all open definitions in the definition workspace.
2. Select File, Rename.

The Rename Definition dialog box appears.

3. Select *Menu* from the Definition Type drop-down list box.
4. Click Rename.
5. From the list of available menus, double-click the menu that you want to rename.
6. Type the new name over the name that is selected on the menu definition.
7. Click Rename.

---

**Note.** If you rename a menu definition, be sure to rename the corresponding registry entries manually.

---

### See Also

[Chapter 2, "Using PeopleSoft Application Designer," Saving Definitions, page 33](#)

## Copying Menu Definitions

Creating a copy of a menu definition is different from renaming a definition. The Save As command creates a copy of the definition with a new name while keeping the old menu definition unchanged.

To copy a menu definition:

1. Open the menu definition that you want to copy.

2. Select File, Save As.

The Save As dialog box appears.

3. Type a new name for the copy of the menu definition.
4. Click OK.

You are prompted to save a copy of any PeopleCode that you have associated with the menu definition.

## Deleting Menu Items

Delete an item to remove it permanently from a menu definition. To help prevent accidental deletions, you can delete only one entry at a time.

To delete a menu item:

1. Select the menu item that you want to delete.
2. Select Edit, Clear.

If you attempt to delete a menu item that is linked to PeopleCode, the system issues a warning.

If you proceed to delete it, the linked PeopleCode menu items are also deleted.

If you delete an item by mistake, before you perform any additional edits or saves, select Edit, Undo to restore the menu item.

---

**Note.** When deleting menu items, remember to delete any corresponding registry entries.

---

## Printing Menu Definitions

In addition to printing a menu definition from the definition workspace, you can obtain a formatted report of all menu definitions by printing the PeopleTools Menu Listing cross-reference report (XRFMENU). This report lists application windows in alphabetical order, and details all menus in each window and all page definitions in each menu. It also includes the associated search record definition name and detail page definition name.

To print a menu definition:

1. Open the menu definition that you want to print so that it appears in the definition workspace as the current definition.
2. Select File, Print.

The standard Windows Print dialog box appears.

## Setting Up Menu Security

Whenever you make a change or add items to an existing custom menu definition, you might need to adjust your security settings. When you add a new menu to a security profile, that menu is available the next time that you sign in to the system. Use the Registration Wizard to assign security.

**See Also**

Chapter 14, "Using the Registration Wizard," page 435

*PeopleTools 8.52: Security Administration, "Understanding PeopleSoft Security"*

## Importing Menu Groups Into Portals

The menu import process creates content references for all components and pages that belong to a menu group. Use this process only to migrate custom menus.

**See Also**

*PeopleTools 8.52: PeopleTools Portal Technologies, "Configuring the Portal Environment," Importing Menu Groups into the Portal Registry*

---

## Defining Pop-up Menus

This section provides an overview of pop-up menus and discusses how to:

- Create pop-up menus.
- Define transfer menu items.

## Understanding Pop-up Menus

Pop-up menus define the transition to another page—either a system-defined page containing a set of standard commands or a pop-up menu page that was created and associated with the pop-up button by an application developer.

You can also use pop-up menus to run a PeopleCode program. PeopleCode in pop-up menus does not share the same limitations as PeopleCode in standard menus, so pop-up transition menus provide an effective alternative to using command push buttons on pages. Pop-up menu PeopleCode programs can be used for any number of purposes, for example:

- To perform a modal transfer.
- To recalculate a field value.
- To trigger a PeopleSoft Workflow business event.

***Behavior of Pop-up Menus in Pages***

These standard icons appear on the page at runtime when a pop-up menu is defined for a page field:



When a user clicks the pop-up icon (PT\_POPUP) on a page, a screen appears with a list of menu items from which to select.



This pop-up icon (PT\_POPUP\_SKIP) appears if a pop-up menu has only one menu item. When a user clicks this icon, the target transaction page appears immediately, skipping an intermediate page containing a list with one menu item.

For example, the Amount field on the following page contains a pop-up icon, which opens the Display in Other Currency page.

Pop-up button on a page

This type of transfer to another component is called a *definitional transfer* to distinguish it from transfers that are made using PeopleCode programs. Definitional transfers are always nonmodal. To run a modal transfer, you must use PeopleCode.

## Creating Pop-up Menus

This section discusses how to:

- Create new pop-up menus.
- Associate pop-up menus with page fields.

### ***Creating New Pop-up Menus***

To create a new pop-up menu:

1. Select File, New.

The New dialog box appears.

2. Select Menu.

3. Click OK.

The New Menu dialog box appears.

4. Select the Popup option for the menu type.

5. Click OK.

A new pop-up menu definition appears.

6. Define the menu items for the pop-up menu.

7. Set the menu properties:

- a. Add a description and comments about your pop-up menu.

- b. On the Use tab, type the menu label that you want to appear for that pop-up.

8. Save the pop-up menu definition.

9. To provide users with access to the pop-up menu, associate it with a page field.

### ***Associating Pop-up Menus With Page Fields***

To associate a pop-up menu with a page field:

1. Open the page definition that contains the page field to which you want to associate the pop-up menu.
2. Right-click the page field and select Page Field Properties.
3. When the properties dialog box for that field appears, select the Use tab.
4. In the Popup Menu group box, select from the available pop-up menus in the drop-down list box.
5. Click OK.
6. Save the page definition.

### ***See Also***

Chapter 13, "Creating Menu Definitions," Setting Menu Item Properties, page 425

## Defining Transfer Menu Items

Transfer menu items, which you create only in pop-up menus, enable you to specify a menu, component, and page to which the user is taken when the transfer menu item is selected. The following procedure describes how to create a new transfer menu item or redefine an existing menu item as a transfer menu item.

To define a transfer menu item:

1. In a pop-up menu definition, double-click a menu item to access its properties.

To create a menu item, double-click the empty rectangle at the bottom of the menu. The Menu Item Properties dialog box for pop-up menus appears.

2. If this is a new menu item, enter a menu item name and label text in the Menu Item fields.

You can create a shortcut key for the menu item by placing an ampersand in the label.

3. Set the transfer properties.

- Click the Transfer button to make this a transfer menu item.
- Click the Define Transfer button to display the Transfer Properties dialog box.

This dialog box enables you to define the properties of a definitional transfer that is made from a pop-up menu item. The component from which you are transferring is the originating component. The component to which you are transferring is the destination component.

## 4. Specify the following:

<b>Portal</b>	Select the navigation registry, such as employee portal.
<b>Node</b>	Select the PeopleSoft database to which you want the transfer menu to point. <i>Local</i> is the default. Valid values include all of the values in the MSGNODENAME column of the PSMSGNODEDEFN table. If you select a node other than <i>Local</i> , the system deselects the Use data from current page in search dialog check box.
<b>Menu</b>	Select the name of the menu containing the destination component.
<b>Component</b>	Select the name of the destination component.
<b>Market</b>	Select the market of the destination component.
<b>Component Item</b>	Select the name of the page to display at the front of the destination component.
<b>Action</b>	<p>Select an action mode (<i>Update/Display</i>, <i>Update/Display All</i>, <i>Correction</i>, or <i>Data Entry</i>) to constrain the user profile to a specific action mode at the time of the transfer. The action modes that are available in the prompt list depend on which modes are specified as valid in the destination component definition.</p> <p>Select <i>Prompt</i> to enable the user profile to select an available action mode at the time of the transfer. The action modes that the user profile can select depend on which action modes are defined as valid in the destination component definition and to which action modes the user profile is granted access in Maintain Security.</p>

## 5. Set the transfer type.

<b>Re-use current window</b>	Select to transfer to the component in the same window in which the transfer is initiated.
<b>Use new window (minimize current window)</b>	Select to minimize the current window and open the destination component in a separate window.

## 6. Specify whether to supply search key values from the originating component.

<b>Use data from current page in search dialog</b>	<p>Select to use the values from the originating component search keys in the search dialog box of the destination component. If these values allow the system to select a unique row from the search record, the search dialog box is bypassed and the destination component opens immediately.</p> <p>Deselect this check box if you want the user to type the search key values into the search dialog box.</p>
--	--

## 7. Click OK to accept the Transfer Properties dialog box settings.

8. Click OK to accept the Menu Item Properties dialog box settings.
9. Save the menu definition.



## Chapter 14

# Using the Registration Wizard

This chapter provides an overview of working with the Registration Wizard and discusses how to:

- Register components and iScripts.
- Register components in multiple portals.
- Register mobile pages.

---

## Understanding the Registration Wizard

This section provides an overview of the Registration Wizard.

### Common Elements Used in This Chapter

<b>content reference</b>	A reference to a URL for a transaction page. Appears as a link in the navigation menu at runtime.
<b>iScript</b>	A specialized PeopleCode function that generates dynamic web content. See <i>PeopleTools 8.52: PeopleCode API Reference</i> , "Internet Script Classes (iScript)."
<b>permission list</b>	A set of access rights to application objects, including processes and reports, web libraries, functions, and other elements. See <i>PeopleTools 8.52: Security Administration</i> , "Setting Up Permission Lists."
<b>portal registry</b>	A tree structure in which content for a portal is organized, classified, and registered. A portal registry consists of folders, content references, and nodes. See <i>PeopleTools 8.52: PeopleTools Portal Technologies</i> , "Understanding Portal Technology," Portal Registry.

## Component Registration

After you create pages and assign them to a component, you must register that component to display the transaction page in the browser. You can use the Registration Wizard to do this. The wizard gathers information, attaches the component to a menu, assigns a permission list to allow security access rights, and places a content reference entry in the portal registry, which displays the component to the user on the navigation menu.

---

**Note.** Before running the Registration Wizard, the menu definition, permission list, and folder must exist. The wizard does not create these definitions. Instead, it matches existing definitions to one another and creates the content reference.

If change control is enabled while you are using the Registration Wizard, you will only be allowed to associate components with locked menus.

---



---

## Registering Components and iScripts

In PeopleSoft Application Designer, open the component definition that you want to register. If you are creating a new component, make sure that you save it before opening the Registration Wizard. For iScripts, open the record definition that holds the iScript function. By convention, such records begin with a WEBLIB\_ prefix.

You can access the Registration Wizard using any of these methods:



Click the Register Component or Register iScript icon in the toolbar.

### Pop-up menu

Select Register Component from the pop-up menu after right-clicking the component.

### Tools menu

Select Register Component or Register iScript from the Tools menu.

---

**Note.** All settings in the Registration Wizard are provided by default from those referenced in the previous wizard session.

---

To register a component or iScript:

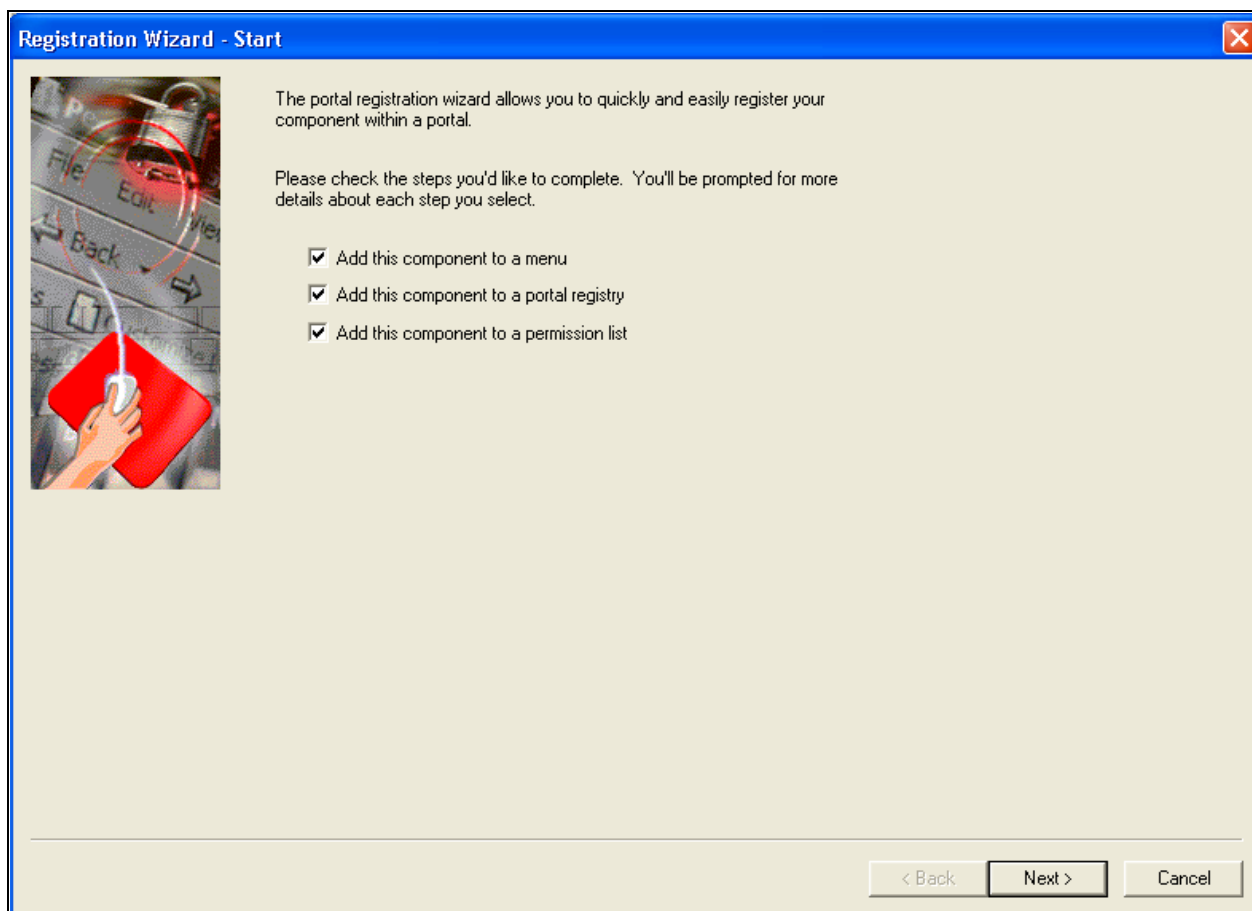
1. Open the component or iScript you want to register.

You can also launch the Registration Wizard from the menu definition by right-clicking a menu item that points to a component.

## 2. Open the Registration Wizard.

Select Tools, Register in Portal or click the icon on the Toolbar.

The Start screen of the Registration Wizard provides the option of adding the component to a menu, a portal registry, a permission list, or a combination of the three. If the component is new and you are running the wizard for the first time, you should select all three options. If you have already run the wizard on this particular component and just need to add it to an additional permission list or portal, deselect the other option, Add this component to a menu.



Registration Wizard - Start dialog box

### **Add this component to a menu**

Select to add the component to a menu. You must create the menu definition first if it does not already exist. You specify the menu on the following screen of the Registration Wizard. If you do not select this option, you will need to add the component to the menu manually using the Menu designer.

See [Chapter 13, "Creating Menu Definitions," Creating Custom Component Menus, page 424.](#)

### **Add this component to a portal registry**

Select if you want to create a content reference for the component and add it to the portal registry you specified on the Create Content Reference screen of the Registration Wizard.

**Add this component to a permission list**

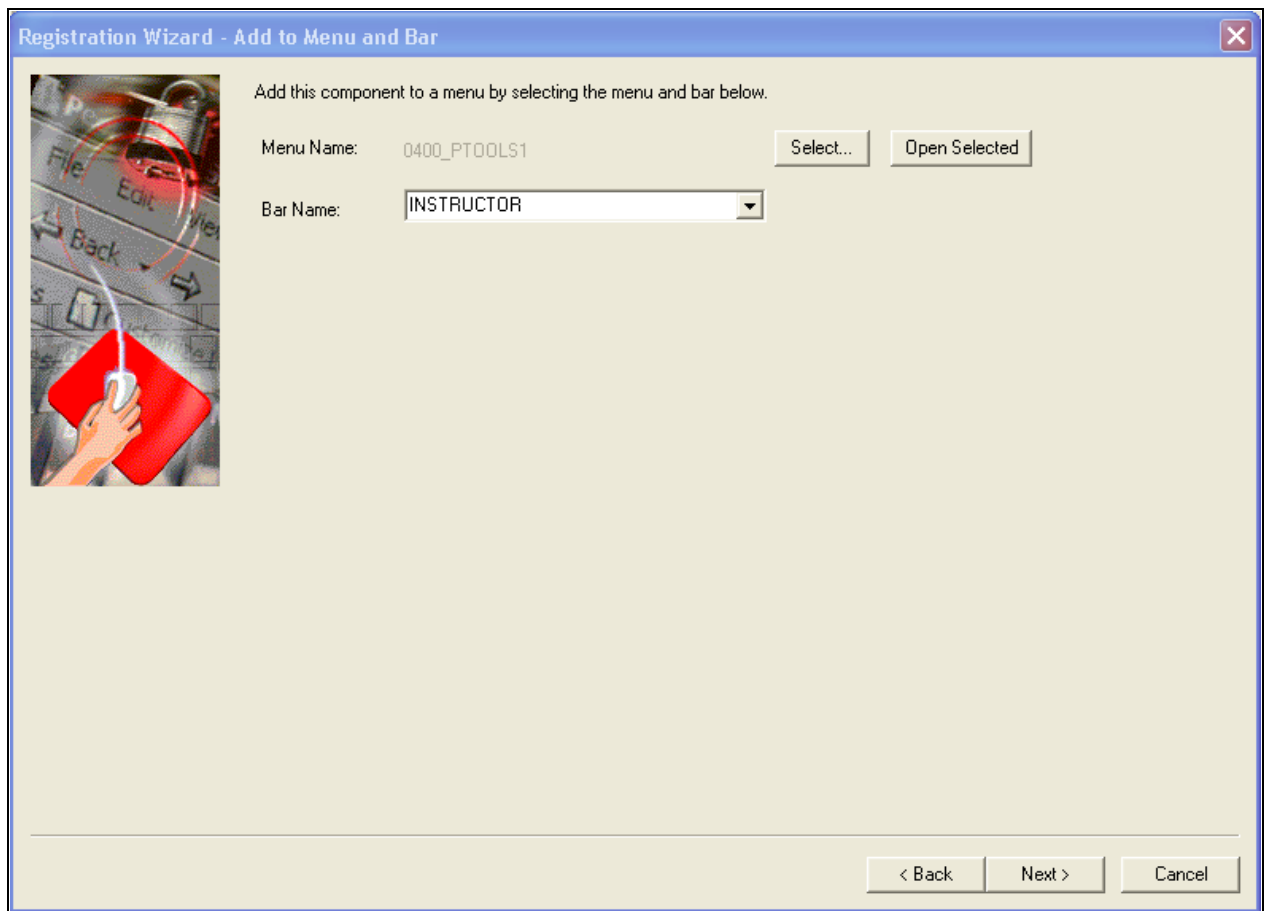
Select if you want to specify the permission list on the final screen of the Registration Wizard.

If the component or iScript is new and you are running the Registration Wizard for the first time, you will probably want to select all three options. If you are rerunning the wizard to add the component to an additional permission list or another portal, you would deselect the other option, , Add this component to a menu.

The wizard remembers the selection that you made the last time that you ran the wizard and selects those values by default.

3. Click the Next button.

If you selected to add the component to a menu, then the next screen, Add to Menu and Bar, opens.

The image shows a Windows-style dialog box titled "Registration Wizard - Add to Menu and Bar". On the left is a graphic of a hand holding a red mouse over a keyboard. The main area contains the text "Add this component to a menu by selecting the menu and bar below." Below this, there is a "Menu Name:" label followed by the text "0400\_PT00LS1" and a "Select..." button. To the right of the "Menu Name" field is an "Open Selected" button. Below the "Menu Name" field is a "Bar Name:" label followed by a drop-down menu showing "INSTRUCTOR". At the bottom of the dialog are three buttons: "< Back", "Next >", and "Cancel".

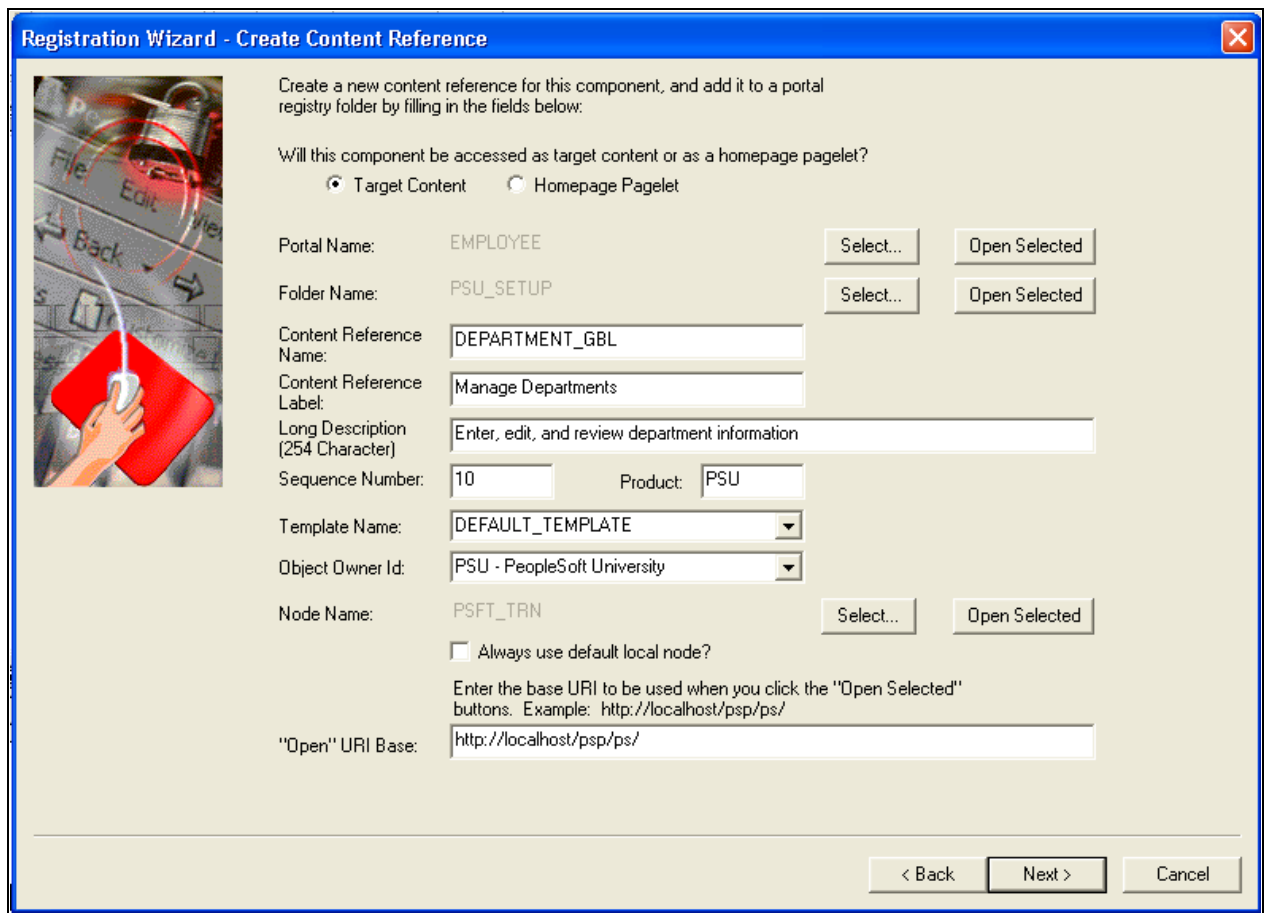
Registration Wizard - Add to Menu and Bar dialog box

4. Click the Select button to find the menu name, click the Bar Name drop-down list box to assign the bar name, and then click the Next button.

These names are not exposed to users, but they are required for internal location purposes. Click the Open Selected button to open the menu that you selected in PeopleSoft Application Designer. To view the menu definition without closing the wizard, drag the Registration Wizard to the side.

- On the Create Content Reference page, select Target Content or Homepage Pagelet.

If the component is a standard transaction, register it as Target Content.



**Registration Wizard - Create Content Reference**

Create a new content reference for this component, and add it to a portal registry folder by filling in the fields below:

Will this component be accessed as target content or as a homepage pagelet?

☒ Target Content    ☐ Homepage Pagelet

Portal Name: EMPLOYEE    Select...    Open Selected

Folder Name: PSU\_SETUP    Select...    Open Selected

Content Reference Name: DEPARTMENT\_GBL

Content Reference Label: Manage Departments

Long Description (254 Character): Enter, edit, and review department information

Sequence Number: 10    Product: PSU

Template Name: DEFAULT\_TEMPLATE

Object Owner Id: PSU - PeopleSoft University

Node Name: PSFT\_TRN    Select...    Open Selected

☐ Always use default local node?

Enter the base URI to be used when you click the "Open Selected" buttons. Example: http://localhost/ps/ps/

"Open" URI Base: http://localhost/ps/ps/

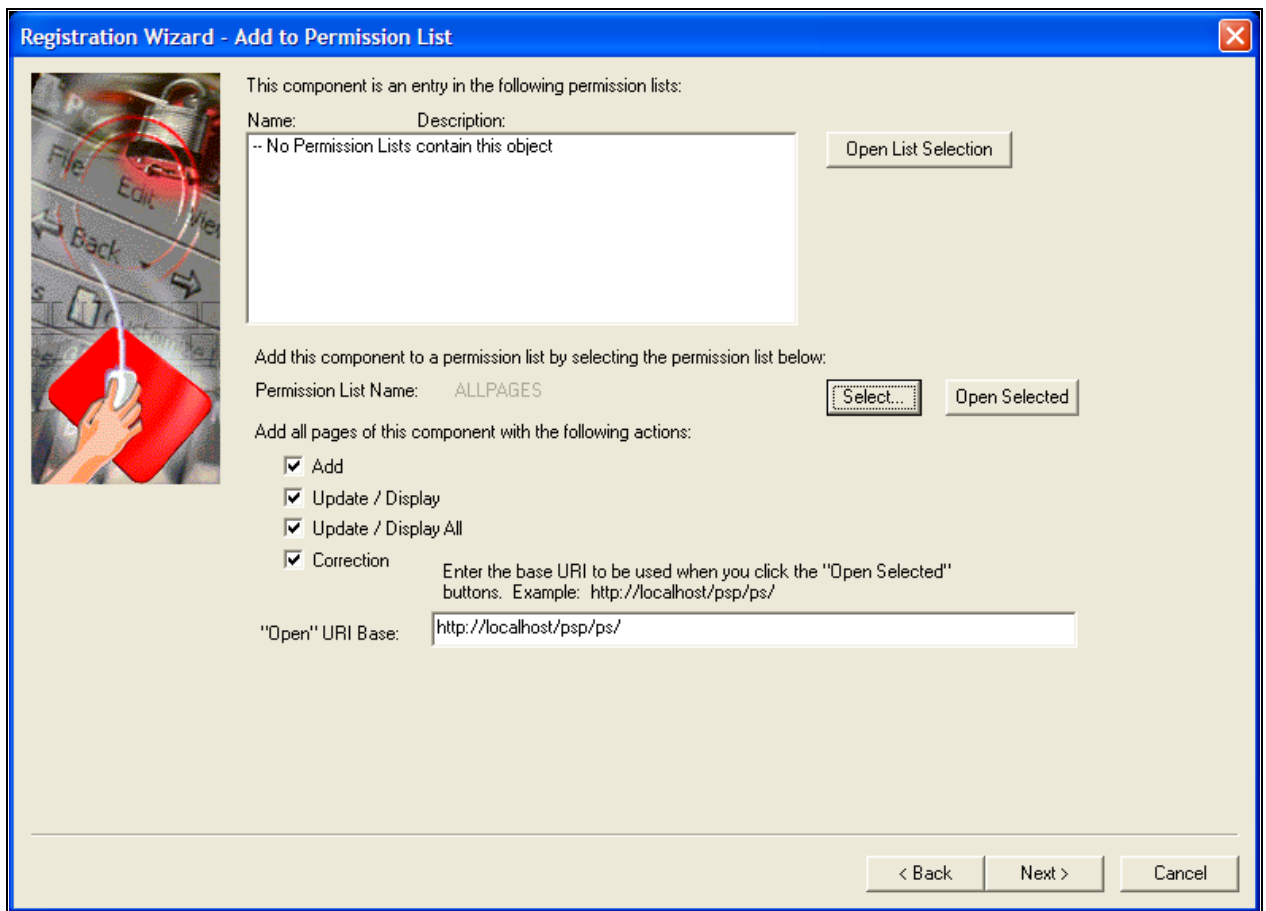
< Back    Next >    Cancel

Registration Wizard - Create Content Reference dialog box

## 6. Complete all fields in the Create Content Reference page.

<b>Target Content</b>	Select this check box to register all forms of target content (excluding the homepage).
<b>Homepage Pagelet</b>	Select to register a new homepage pagelet. The system applies the homepage template automatically.
<b>Portal Name and Folder Name</b>	Click the Select button to change the portal or folder name. Click the Open Selected button to launch a browser that will take you to the administration page to create a new folder, if necessary. The default values are the last ones referenced by the wizard.
<b>Content Reference Name</b>	The name is automatically generated, including the market code as the suffix.
<b>Content Reference Label</b>	Enter the name of the CREF hyperlink that the user sees as the registry entry.
<b>Long Description</b>	Enter hover text for the CREF link.
<b>Sequence Number</b>	Enter a numeral to control the order in which the component appears within the chosen folder.
<b>Product</b>	Enter a two-character product code.
<b>Template Name</b>	Specify the template to be used from the drop-down menu.  If you specified that the target content should be accessed as a homepage pagelet, then this field will not appear and the system will apply the homepage template automatically.
<b>Object Owner Id</b>	Select the application development group that owns the object.
<b>Node Name</b>	Click Select to change the Node name, or click the Open Selected button to launch the administration page to view the properties of the node.
<b>Always use default local node</b>	Select to always override the selected node name with the specified default local node.
<b>"Open" URI Base</b>	Enter the Universal Resource Identifier (URI) that specifies for PeopleSoft Application Designer the location of your PeopleSoft Pure Internet Architecture site. This entry is for informational purposes only. It is used only for the Open Selected button and not stored to the database.  If you entered a base URI on the Browser tab of the Options dialog box, that value appears in this field.  The URI you enter in this field is reflected on the remaining pages of the Registration Wizard. If you make a change to this field on any page, the change is reflected throughout.

7. Click the Next button.
8. You see the Add to Permission List page.



**Registration Wizard - Add to Permission List**

This component is an entry in the following permission lists:

Name:	Description:
-- No Permission Lists contain this object	

Open List Selection

Add this component to a permission list by selecting the permission list below:

Permission List Name: ALLPAGES Select... Open Selected

Add all pages of this component with the following actions:

- ☒ Add
- ☒ Update / Display
- ☒ Update / Display All
- ☒ Correction

Enter the base URI to be used when you click the "Open Selected" buttons. Example: <http://localhost/psp/ps/>

"Open" URI Base:

< Back    Next >    Cancel

Registration Wizard - Add to Permission List dialog box

Click the Select button to find the menu name, and then use the drop-down list box to select the name that appears in the bar.

Menus are used as logical groupings to which you can apply security. PeopleSoft components are addressed by menu name (the URL includes the menu name). To add this component to a menu, specify the menu name and bar name.

If you did not select the Add this component to a menu check box on the Prompt page, this page displays the menu name and the Select button only. You then must select the correct menu that already contains the component.

If you are working with an iScript, you must select the iScript function to register.

9. Click the Next button.
10. Select Target Content if your component is a standard transaction and complete the appropriate fields.

The previous term table defines each of these fields.

If you did not select Add this Component to a Portal Registry on the Prompt page, this page does not appear.

11. Click the Next button.
12. Select the appropriate permission list, select all the actions that you want to grant this component or iScript, and enter the base URI.

Use the Select button to view and select a permission list.

If you did not select Add this Component to a Permission List, this page does not appear.

13. Click the Next button.
14. Use the Finish page to review the selections made so far. If you need to change anything, use the Back button to edit your entries.

Select the options in the Add to project area to place your item into an active PeopleTools project. This is cumulative behavior: the more times you run the wizard while the project is active, the more menus, registry structures, and permission lists are added to that project.

15. Click Finish to complete the registry process into the specified portal.

If you click Cancel all of the entries that you have made in the wizard will be lost and no changes will be made.

### **See Also**

[Chapter 13, "Creating Menu Definitions," page 423](#)

*PeopleTools 8.52: Security Administration, "Setting Up Permission Lists"*

---

## **Registering Components in Multiple Portals**

After you use the Registration Wizard to register your new component into the portal, you may need to register the component in other portals also.

You can do this in two ways.

- You can use the Portal Registration Wizard again. Select only the second option, Add this component to a portal registry. Then reenter the content reference label, long description, and sequence number (if not 1) to match the entry of the other portal.
- You can copy the component to other portals from portal Administration pages in the web client.

See *PeopleTools 8.52: PeopleTools Portal Technologies*.

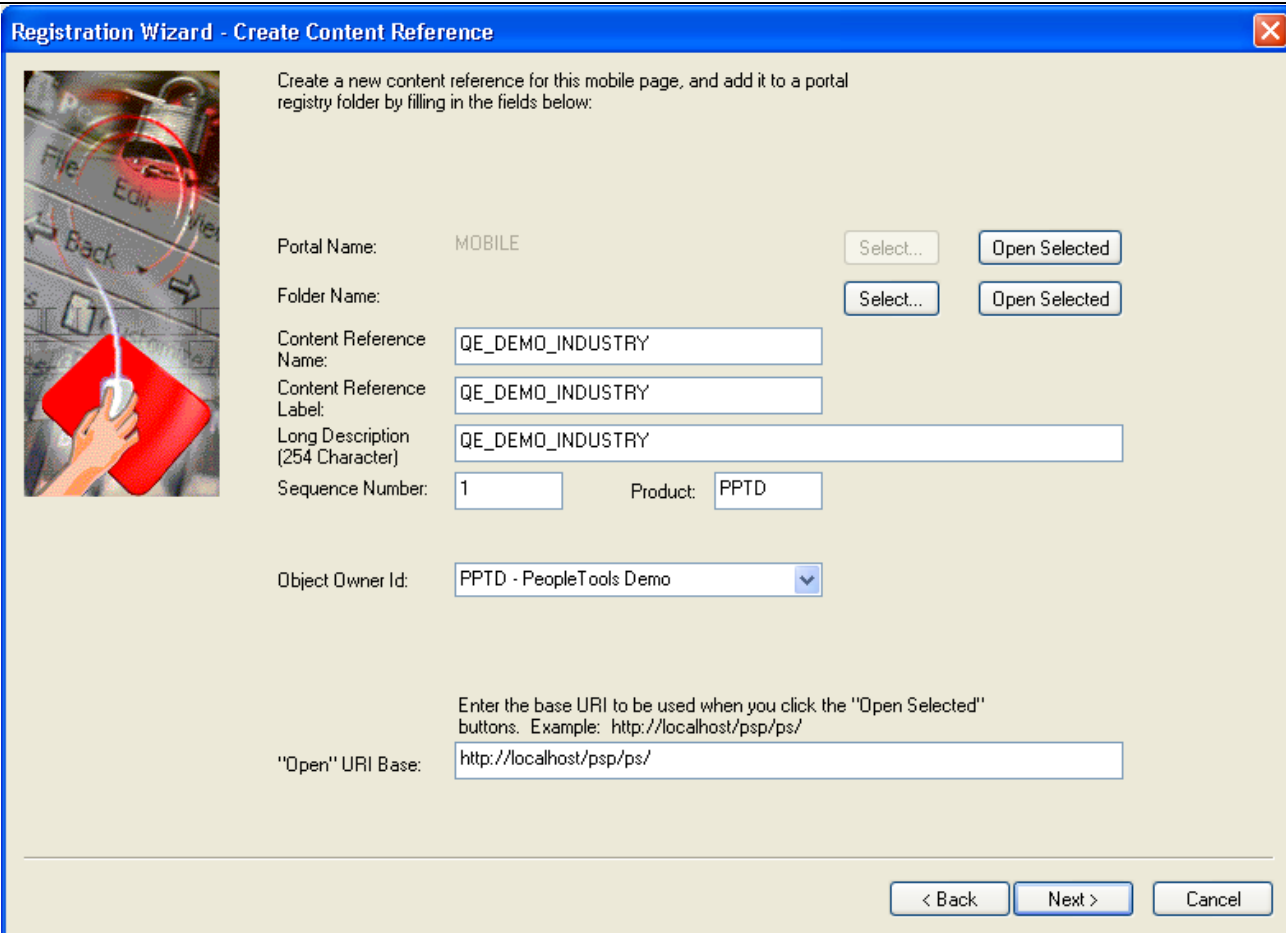
---

## **Registering Mobile Pages**

As with a component or iScript, you can also register a mobile page in the portal using the Registration Wizard, though the wizard is slightly different. This wizard always assigns the mobile page to the MOBILE portal and does not require a node name.



**Important!** PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.



**Registration Wizard - Create Content Reference**

Create a new content reference for this mobile page, and add it to a portal registry folder by filling in the fields below:

Portal Name: MOBILE Select... Open Selected

Folder Name: Select... Open Selected

Content Reference Name: QE\_DEMO\_INDUSTRY

Content Reference Label: QE\_DEMO\_INDUSTRY

Long Description (254 Character): QE\_DEMO\_INDUSTRY

Sequence Number: 1 Product: PPTD

Object Owner Id: PPTD - PeopleTools Demo

Enter the base URI to be used when you click the "Open Selected" buttons. Example: http://localhost/ps/ps/  
 "Open" URI Base: http://localhost/ps/ps/

< Back Next > Cancel

#### Registration Wizard for mobile dialog box

To register a mobile page in the portal:

1. Open the mobile page that you want to register.
2. Open the Registration Wizard.

Use any one of three methods to do this:

- Click the Registration Wizard icon in the toolbar.
  - Select Tools, Register Mobile Page from the menu.
  - Select Register from the pop-up menu.
3. Set the Folder Name and content reference information.
  4. Assign the Sequence Number and Product.
  5. Select the object owner ID from the drop-down list box.

6. Click Next.

The final screen of the wizard opens, enabling you to verify the settings that you selected. The Registration Wizard also adds your registry entry to the current project. The Registry entry check box is selected by default.

7. Review your settings in the output window and click Finish.

## Chapter 15

# Creating Style Sheet Definitions

This chapter provides an overview of style sheets and style classes and discusses how to:

- Create standard and sub style sheets.
- Set style class attributes.
- Work with free form sub style sheets.
- Work with default style sheets.
- Override default page style properties.
- Modify page tabs.
- Modify push buttons in SWAN style sheets.

---

## Understanding Style Sheets and Style Classes

This section discusses:

- Overview of style sheets and style classes.
- Style sheet types.
- Style classes.

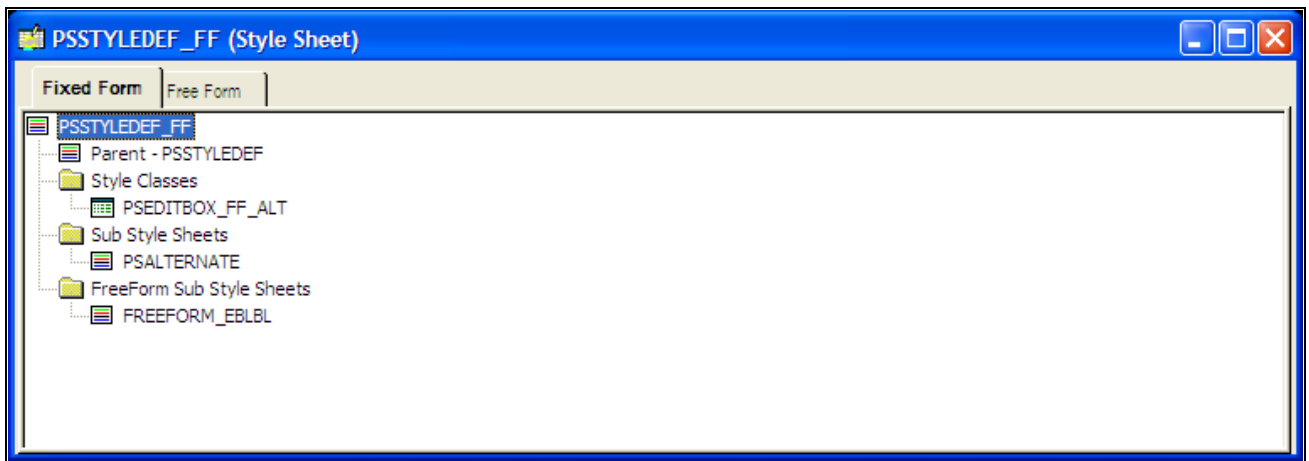
## Overview of Style Sheets and Style Classes

Style sheets are definitions that store margins, spacing, fonts, borders, color and other layout settings for particular elements of a page. When you use a style sheet, the system applies the predefined layout settings or style classes to every page in an application. By storing all page layout settings in a style sheet definition, you eliminate the need to redefine the same settings for each page separately, you can quickly reconfigure and apply new settings, and you maintain a consistent appearance throughout the application.

In PeopleSoft applications, a style sheet is a definition, like a record or page definition, that you create and maintain in PeopleSoft Application Designer. A style sheet is a standalone definition that you can upgrade, delete, rename, and so on.

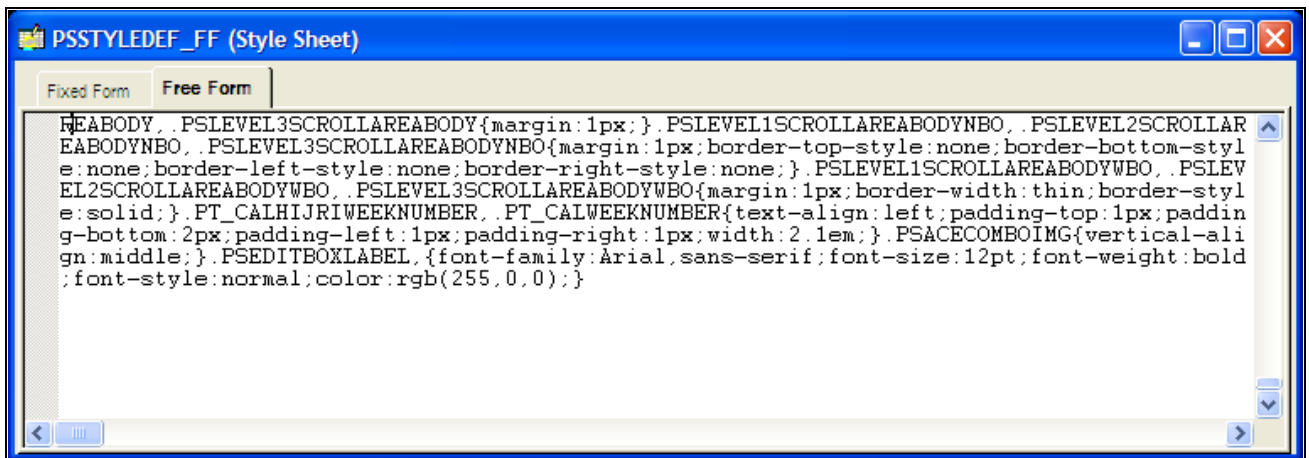
You define style sheets by using the Fixed Form and Free Form tabs.

In this example, you see the Fixed Form tab, which you access to insert style sheets, sub style sheets, free form sub style sheets, or style classes into a definition:



Style sheet definition showing the Fixed Form tab

In this example, you see the Free Form tab, which you access to enter text when you create a free form sub style sheet. For standard style sheets and sub style sheets, use this view to examine or copy the text that is generated by the style sheet definition:



Style sheet definition showing the Free Form tab

---

**Note.** For standard style sheets and sub style sheets, the Free Form tab text is read-only.

---

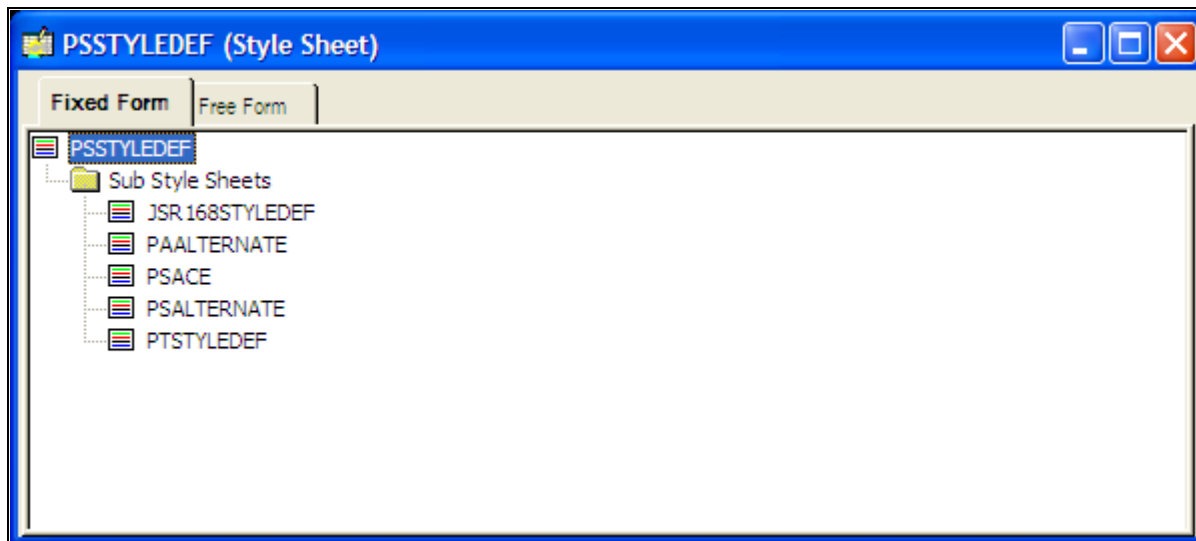
## Style Sheet Types

You can create three style sheet types using PeopleSoft Application Designer:

- Standard style sheet
- Sub style sheet
- Free form sub style sheet

## **Standard Style Sheet**

This is an example of a standard style sheet:



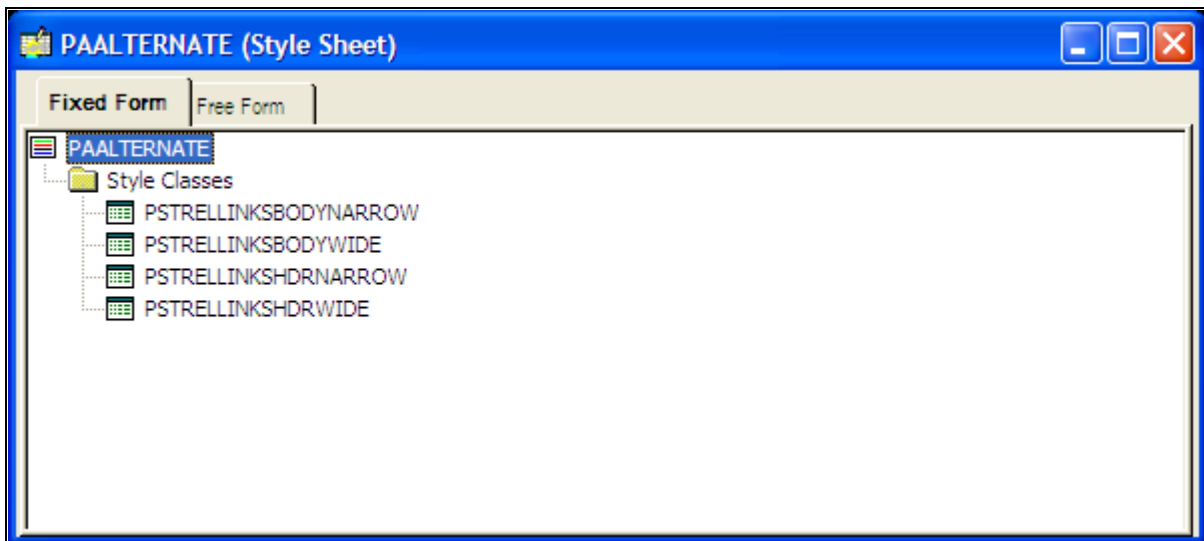
Example of a standard style sheet

Standard style sheets are the primary organizers for working with style sheets and style classes. Standard style sheets are containers that can include sub style sheets, free form sub style sheets, and style classes in any combination; however, standard style sheets cannot contain other standard style sheets.

Standard style sheets can inherit style classes from other standard style sheet definitions through the Parent Style Sheet property. The parent style sheet propagates all of its style classes to the child style sheet. Only standard style sheets can have parent style sheets and be used as parent style sheets.

## **Sub Style Sheet**

This is an example of a sub style sheet:



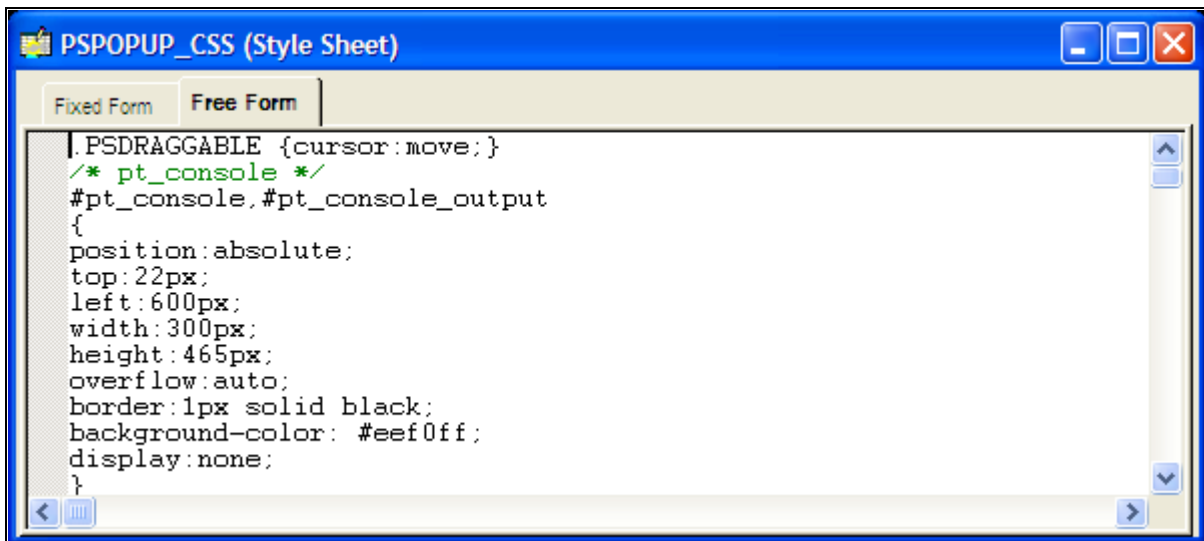
Example of a sub style sheet

In the same way that PeopleTools supports subrecords, you can define sub style sheets. The primary purpose of sub style sheets is to organize and reuse style classes. You can insert sub style sheets into standard style sheets as well as into other sub style sheets. Sub style sheets neither have parent style sheets nor can be used as parent style sheets.

To use sub style sheets, you must insert them into standard style sheets.

### ***Free Form Sub Style Sheet***

This is an example of a free form sub style sheet:



Example of a free form sub style sheet

Free form sub style sheets are text-based sub style sheets that enable you to take advantage of Cascading Style Sheets - Level 2 (CSS2), AJAX, and DHTML features. With a free form sub style sheet, you can create the style sheet in a third-party editor and then copy the style sheet text into the Free Form tab of the free form sub style sheet. On free form sub style sheets, the Fixed Form tab is grayed out because free form sub style sheets do not support inheritance or aggregation. Free form sub style sheets cannot have parent style sheets nor can they be used as parent style sheets.

Note these important points regarding the use of free form sub style sheets:

- Like other sub style sheets, to use free form sub style sheets you must insert them into a standard style sheet or into a sub style sheet that is part of a standard style sheet.
- When creating free form style sheets, style class names must be identical to the PeopleTools default style class names.

If you replace the default style sheet or override the page style sheet with a style sheet that includes only free form text, the free form sub style sheet must define and include *all* default style classes used by the application. Unexpected results will occur in both PeopleSoft Application Designer and PeopleSoft Internet Architecture pages if a default style class is missing.

See [Chapter 15, "Creating Style Sheet Definitions," Default Style Classes, page 450](#).

- At runtime, style classes in free form sub style sheets take precedence over PeopleTools-formatted style classes of the same name.

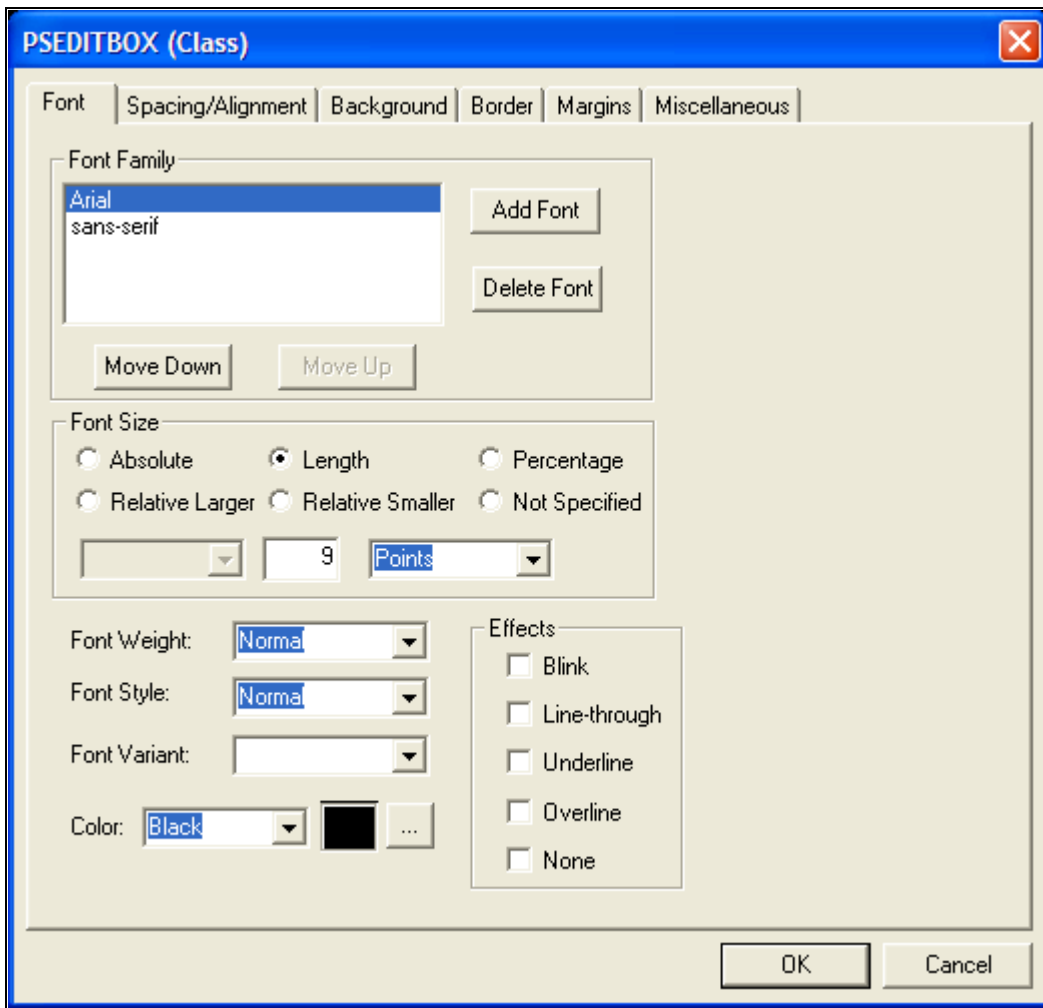
---

**Note.** Free form sub style sheets are supported in PeopleTools release 8.50 and later. If you attempt to import a free form sub style sheet into an earlier release, the free form sub style sheet imports as an empty standard style sheet.

---

## Style Classes

Access a style class dialog box.



PSEDITBOX style class properties dialog box

Standard style sheets and sub style sheets comprise standard style classes. The style class determines the visual properties of pages and page controls by controlling a broad range of characteristics, including font size, spacing, alignment, border width, font weight, and color.

Style classes in different style sheets are independent of each other. You can copy existing style classes and paste, or insert, them into any style sheet. However, the most efficient way to share style classes across multiple style sheets is to group style classes into sub style sheets.

### **Default Style Classes**

Default style classes are the PeopleTools style classes that the system explicitly associates with a page element, such as the page, each page control type, page control labels, and so on. PeopleSoft applications use the default style classes to render the page and all page elements at design time and at runtime, unless you specify a style property override.

See [Chapter 15, "Creating Style Sheet Definitions," Overriding Default Page Style Properties, page 479.](#)

Note these points about default style classes:

- If you change an attribute of a default style class, the change affects *every* occurrence of the page element throughout an application, except the instances that do not use the default style.



- The application style sheet (or any page level override) must include *all* default style classes for the system to render pages correctly at design time and at runtime. The default style classes can be accounted for in standard or free form format in any combination. However, unexpected results will occur in both PeopleSoft Application Designer and PeopleSoft Internet Architecture pages if a default style class is missing.

---

**Note.** PeopleTools style sheets include all default style classes. You should not modify any PeopleTools style sheets or the attributes of default style classes.

---

Most default style class names enable you to easily identify the page element with which they are associated. For example, PSEDITBOX is the name of the default style class for all edit box page controls. When you examine a specific edit box control, the value in the Style field, however, is not the *name* of the default style class. The Style field value of *\*\*\*Use Default Value\*\*\** identifies any page element using its default style class.

This PeopleBook includes an appendix that lists and describes the default style classes.

See [Appendix C, "Using Default Style Classes," Default Style Classes, page 591](#).

### **Custom Style Classes**

This list includes some examples of custom style classes:

- Creating new PeopleTools style classes, inserting them into standard style sheets, and using them as style overrides to create unique pages in an application.
- Creating and inserting override style classes that supplant style classes inherited from a parent style sheet.
- Overriding page style classes with non-default, existing PeopleTools style classes.
- Creating free form style sheets that reconfigure one or more default style classes.

---

## **Creating Standard and Sub Style Sheets**

This section discusses how to:

- Create standard style sheets or sub style sheets.
- Rename standard style sheets or sub style sheets.
- Delete standard style sheets or sub style sheets.
- Insert sub style sheets into style sheets.
- Insert style classes into style sheets.
- Inserting override style classes into style sheets.

### **Creating Standard Style Sheets or Sub Style Sheets**

To create a new standard style sheet or sub style sheet:

1. Select File, New.
2. Select *Style Sheet* from the Definition drop-down list box and click the OK button.
3. Select the style sheet type.
4. Click the OK button.
5. Select File, Definition Properties.
6. Select the style sheet type and parent style sheet.

The Parent Style Sheet field is enabled if you create a standard style sheet only. The default parent style sheet is the application style sheet that you set on the PeopleTools Options page.

7. Enter a description.
8. Save the style sheet.

## Renaming Standard Style Sheets or Sub Style Sheets

To rename standard or sub style sheets:

1. Select File, Rename.
2. Select *Style Sheet* in the Definition drop-down list box and then enter search criteria to find the style sheet definition that you are deleting.
3. Click the Rename button; the definitions that match the search criteria appear in the list box.
4. Double-click the style sheet definition that you want to rename.
5. Enter the new name of the style sheet definition in the highlighted area and press the Enter key.
6. Click the OK button to confirm the rename.

---

**Note.** When you rename a standard or sub style sheet, PeopleSoft Application Designer automatically renames all references to the style sheet.

---

## Deleting Standard Style Sheets or Sub Style Sheets

To delete standard or sub style sheets:

1. Select File, Delete.
2. Select *Style Sheet* in the Definition drop-down list box and then enter search criteria to find the style sheet definition that you want to delete.
3. Press the Delete key; the definitions that match the search criteria appear in the list box.
4. Click the style sheet that you want to delete and press the Delete key.
5. Click the OK button to confirm the deletion or click the Cancel button to cancel the operation.

---

**Note.** When you delete a style sheet, PeopleSoft Application Designer automatically deletes all references to the deleted style sheet. If you delete a style sheet that contains sub style sheets, the sub style sheets are *not* deleted; they become independent definitions.

---

## Inserting Sub Style Sheets into Style Sheets

You can insert a sub style sheet into a standard style sheet or another sub style sheet.

To insert a sub style sheet into style sheets:

1. Create or open the receiving style sheet.
2. Select Insert, Insert Sub Style Sheet.
3. In the drop-down list boxes, select the sub style sheet type and the sub style sheet that you want to insert.
4. Click the OK button.
5. Save the sub style sheet.

You can remove sub style sheets from within other style sheets by highlighting the sub style sheet and pressing the Delete key. This action does *not* delete the definition from the database; it removes the sub style sheet from the current definition only.

## Inserting Style Classes into Style Sheets

You can insert style classes into any standard style sheet or sub style sheet.

To insert a style class into a style sheet:

1. Open the style sheet in which you want to create the new style class.
2. Select Insert, Insert Style Class.
3. Enter the name of the new class.
4. Copy attributes from another style class by selecting the style class name from the Copy Values From Style Class drop-down list box.

Leave the Copy Values From Style Class field empty if you want to set all style class properties manually.

5. Click the OK button.
6. Configure style class attributes, as necessary.

See [Chapter 15, "Creating Style Sheet Definitions," Setting Style Class Attributes, page 454.](#)

You can remove a style class from a style sheet by highlighting the class and pressing the Delete key. This action *does* delete this instance of the style class from the database.

---

**Note.** For efficiency, clone style classes from other style sheets. Then, either use the Insert menu or paste the style class into the new style sheet. At that time, the system enables you to enter the new style class name. Be aware that when copying style classes, the copy or clone will not contain any of the related language data associated with the original.

---

## Inserting Override Style Classes into Style Sheets

When you create a style sheet and configure the Parent Style Sheet property, the new style sheet inherits all of the style classes of the parent. If you want to change the attributes of an inherited default style class in the new style sheet, you insert an override style class. To insert an override style class into a style sheet:

1. Open the style sheet.
2. Verify that the Parent Style Sheet property is configured correctly.
3. Select Insert, Override Style Class.
4. Select the style class name from the Style Class drop-down list box and click the OK button.

The class properties dialog box appears showing the old settings.

5. Configure the override style class, as necessary.

See [Chapter 15, "Creating Style Sheet Definitions," Setting Style Class Attributes, page 454.](#)

6. Click the OK button.
7. Save the style sheet definition.

---

## Setting Style Class Attributes

This section provides overviews of class attributes and fonts and discusses how to:

- Access class attributes.
- Specify fonts.
- Set font attributes for a specific language.
- Specify spacing and alignment.
- Specify background attributes.
- Reference a background image URL.
- Specify border attributes.
- Specify margins.
- Specify miscellaneous attributes.

## Understanding Class Attributes

Class attributes are grouped into categories and appear in a tabbed dialog box for each class.

---

**Note.** PeopleSoft Application Designer does not render certain style class formatting attributes at design time. You might be able to see the attribute at runtime only. Also, not all browsers support every style class attribute.

---

PeopleSoft Pure Internet Architecture supports the attributes that are listed in the World Wide Web Consortium (W3), Cascading Style Sheets, Level 1.

See <http://www.w3.org/TR/REC-CSS1>.

Every class has each of these major attributes, which correspond to a tab in the Class dialog box.

<b>Font</b>	Controls the font in which text appears. If more than one font is specified, the browser displays the first font; if it cannot find it, it displays the next font.
<b>Spacing/Alignment</b>	Controls the spacing in addition to the default spacing. You can specify additional spacing between words and between letters. You can also specify the alignment, height, indent, and white space.
<b>Background</b>	Controls the background colors for the page control. If the background is an image, you can select how the image appears.
<b>Border</b>	Controls the border that appears around the element. You can specify width, color, and style.
<b>Margins</b>	Controls the margins and padding on each side of the element.
<b>Miscellaneous</b>	Controls the display of list-item markers, URLs, and cursor formats.

## Understanding Generic and Installed Fonts

When you add a font, the dialog box provides a choice between generic fonts and installed fonts.

Generic fonts can appear in all browsers. This table lists examples of some of the generic fonts:

<b><i>Generic Font</i></b>	<b><i>Example</i></b>
Cursive	Zapf-Chancery
Fantasy	Western
Monospace	Courier

<b><i>Generic Font</i></b>	<b><i>Example</i></b>
Sans serif	Helvetica
Serif	Times New Roman

### ***Installed Fonts***

Installed fonts are installed on your computer. If you use an installed font for a style class, you should:

- Verify that *all* users of your application have the same installed font.
- Specify the installed font first, followed by a generic font.

If the installed font is not available to users of your application, they can use the generic font.

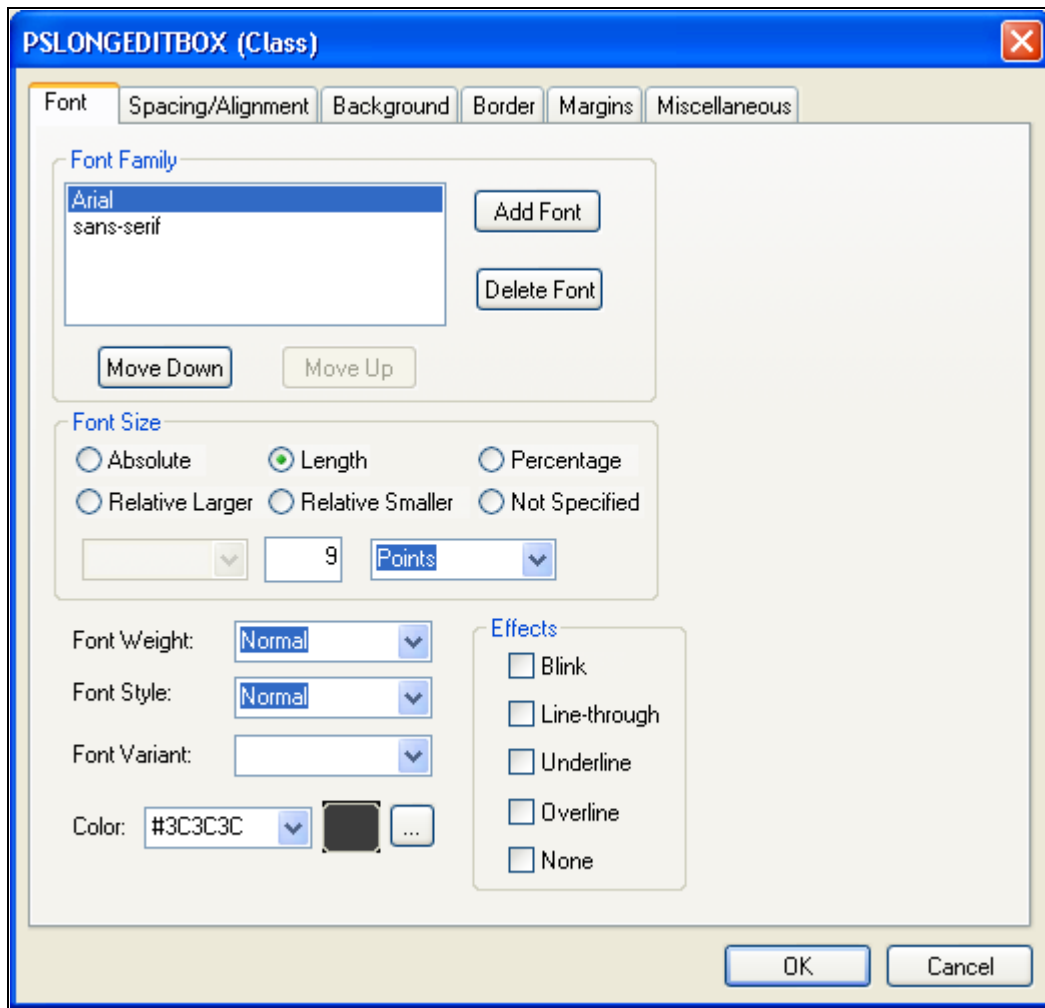
## **Accessing Class Attributes**

To access class attributes:

1. Open a style sheet.
2. Double-click the class name.

## **Specifying Fonts**

Access the class dialog box and select the Font tab.



PSLONGEDITBOX (Class) dialog box: Font tab

### Font Family

Specify the font in which you want the text to appear. You can specify more than one font so that if the browser cannot display the first font, it will attempt to display the next font, and so on, until it finds a font that it can display.

### Font Size

#### Absolute

Select to use a size from the drop-down list box (*xx-small*, *x-small*, *small*, *medium*, and so on). Adjacent sizes have a scaling factor of 1.5 between them.

#### Length

Specify the size with a number and select the measurement type from the drop-down list box (*em*, *ex*, *Pixels*, *Inches*, *Millimeters*, and so on).

#### Percentage

Specify a number, followed by a percent sign (%). Percentage values are always relative to the parent element.

<b>Relative Larger and Relative Smaller</b>	Select to interpret the font relative to the table of font sizes and the font size of the parent element. For example, if the parent element has a font size of medium and you select Relative Larger, this font size is large.
<b>Font Weight</b>	Select the weight of the font. The values <i>100</i> to <i>900</i> form an ordered sequence in which each number indicates a weight that is at least as dark as its predecessor. The value <i>Normal</i> is synonymous with <i>400</i> , and <i>Bold</i> is synonymous with <i>700</i> .
<b>Font Style</b>	Select the style of the font. Values are: <i>Normal</i> , <i>Italic</i> , or <i>Oblique</i> . These values are matched to existing fonts with those names.
<b>Font Variant</b>	Select a variant. Values are: <i>Normal</i> or <i>Small Caps</i> . In a small caps font, the lowercase letters look similar to the uppercase ones, but they are a smaller size and have slightly different proportions.
<b>Color</b>	Specify a color or select a color from a color palette.

## Setting Font Attributes for a Specific Language

You can change the fonts that are available based on the base language settings. The default for the base language is English, providing generic fonts. If these fonts are not appropriate for a particular language, you must set the language preference before opening a style sheet. You can then set the font name list, font size, or font weight for the selected language.

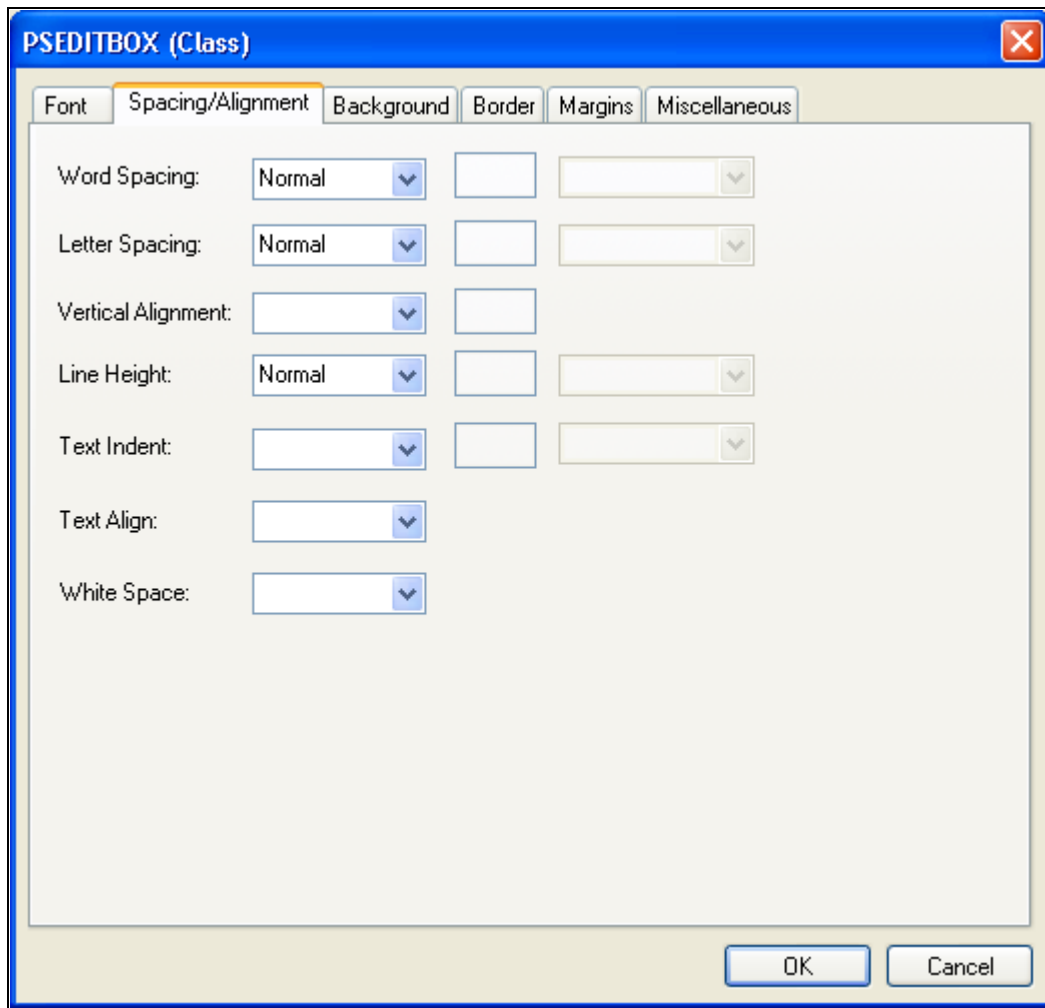
To set font attributes for a specific language:

1. Select PeopleTools, Utilities, International, Preferences from your browser.
2. Select the appropriate language for the style sheet from the drop-down list box.
3. Save the new selected language as the base language.
4. Open the style sheet definition in which you want to save the font name list, font size, or font weight for the selected language.

## Specifying Spacing and Alignment

Access the class dialog box and select the Spacing/Alignment tab.





PSEDITBOX (Class) dialog box: Spacing/Alignment tab

### **Word Spacing and Letter Spacing**

Select *Normal* or *Length*. If you select *Length*, specify a number to indicate an addition to the default space between words or letters. Values can be negative, but implementation-specific limits might exist.

**Vertical Alignment**

Select the vertical positioning of the element. Values are:

*Baseline:* Aligns the baseline (or the bottom if the element does not have a baseline) of the element with the baseline of the parent.

*Sub:* (subscript) Aligns the top of the element with the baseline of the parent element.

*Super:* (superscript) Aligns the baseline of the element with the top of the parent element.

*Top:* Aligns the top of the element with the tallest element in the line.

*Text Top:* Aligns the top of the element with the top of the parent elements font.

*Middle:* Aligns the vertical midpoint of the element (typically an image) with the baseline plus half of the *x*-height of the parent.

*Bottom:* Aligns the bottom of the element with the lowest element on the line.

*Text Bottom:* Aligns the bottom of the element with the bottom of the parent elements font.

*Percentage:* Refers to the value of the line-height property of the element itself. It raises the baseline (or the bottom if it has no baseline) of the element to the specified amount above the baseline of the parent. Negative values are possible. For example, a value of *-100%* lowers the element so that the baseline of the element is where the baseline of the next line should have been. This setting allows precise control over the vertical position of elements (such as images that are used in place of letters) that do not have a baseline.

---

**Note.** Using the top and bottom alignments could cause a loop in which element dependencies occur.

---

**Line Height**

Set the distance between the baselines of two adjacent lines. When a numerical value is specified, the line height is provided by the font size of the current element multiplied with the numerical value. This differs from a percentage value in the way it inherits. When a numerical value is specified, child elements inherit the factor itself, not the resultant value (as is the case with percentage and other units). Negative values are not allowed.

**Text Indent**

Specify the indentation that appears before the first formatted line. This value can be negative, but implementation-specific limits might exist. An indentation is not inserted in the middle of an element that was broken by another.

**Text Align**

Specify how text is aligned in the element. Because Text Align inherits, all block-level elements inside the DIV element with CLASS=center are centered.

---

**Note.** Alignments are relative to the width of the element, not the canvas.

---

**White Space**

Specify how to handle white space inside the element. Values are:

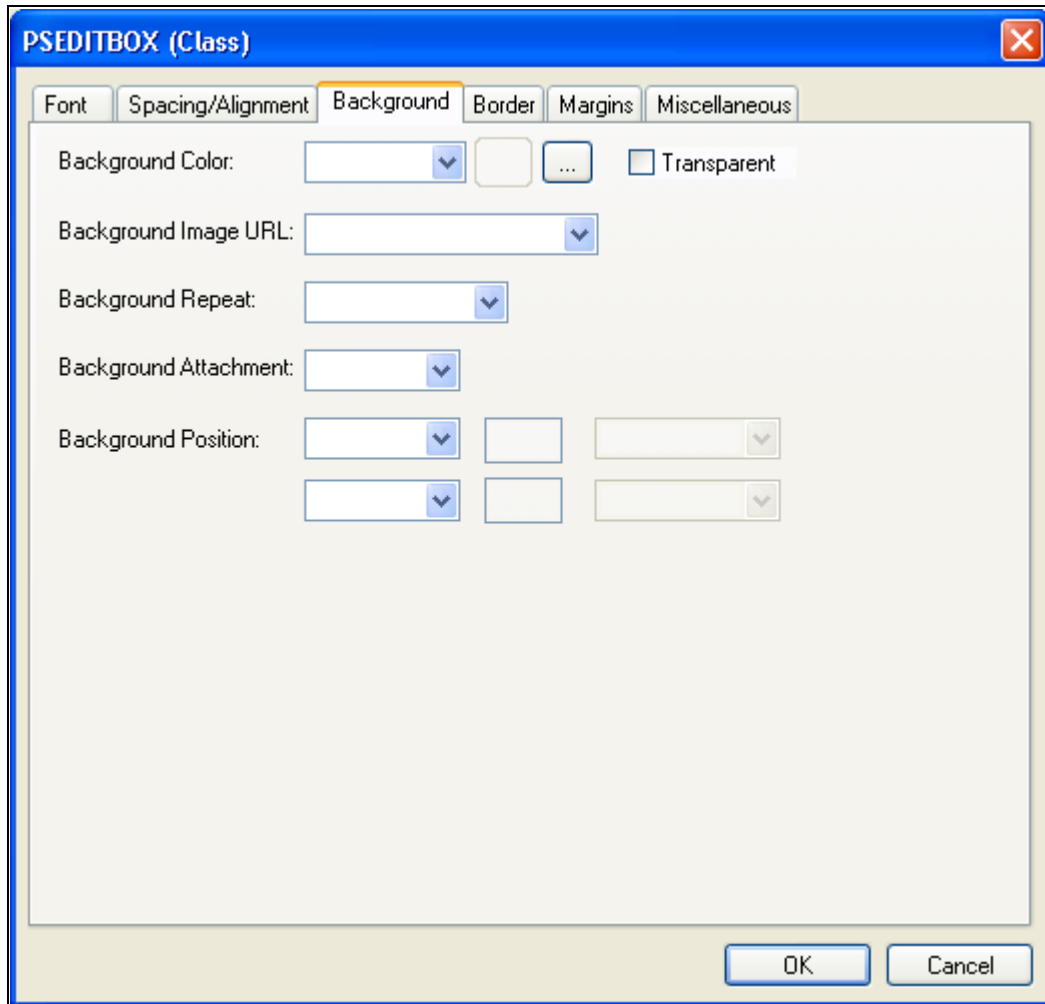
*Normal*: White space is collapsed.

*Pre*: Behaves like the PRE element in HTML.

*Nowrap*: Wrapping is done only through elements with a line break.

**Specifying Background Attributes**

Access the class dialog box and select the Background tab.



PSEDITBOX (Class) dialog box: Background tab

**Background Color**

Specify the background color of an element.

<b>Background Image URL</b>	<p>Set the background image of an element to an image file that was previously defined. When you set a background image, also set a background color to use when the image is unavailable. When the image is available, it appears on top of the background color.</p> <p>See <a href="#">Chapter 15, "Creating Style Sheet Definitions," Referencing a Background Image URL, page 464.</a></p>
<b>Background Repeat</b>	If you specify a background image, indicate whether the image is repeated and, if it is, how many times.
<b>Background Attachment</b>	If you specify a background image, indicate whether it is fixed to the canvas or whether it scrolls along with the content.
<b>Background Position</b>	<p>If you specify a background image, set its initial position. Values are:</p> <p><i>Top:</i> Aligns the background to the top of the element.</p> <p><i>Left:</i> Aligns the background to the left of the element.</p> <p><i>Center:</i> Aligns the background to the midpoint of the element (typically an image).</p> <p><i>Bottom:</i> Aligns the background to the bottom of the element.</p> <p><i>Right:</i> Aligns the background to the right of the element.</p> <p><i>Length:</i> Specifies the length of the background and the unit of measure.</p> <p><i>Percentage:</i> Refers to the value of the line-height property of the background itself. It raises the baseline of the element (or the bottom, if it has no baseline) the specified amount above the baseline of the parent. Negative values are possible. For example, a value of <i>-100%</i> lowers the element so that the baseline of the element is where the baseline of the next line should have been. This setting allows precise control over the vertical position of elements without a baseline, such as images used in place of letters.</p>

### **Examples of Percentage and Length Value Pairs and Keywords**

This table lists background positions, value pairs, and their effect:

<b>Background Position</b>	<b>Value Pair</b>	<b>Effect</b>
Percentage	0 and 0	Places the upper, left-hand corner of the image in the upper, left-hand corner of the box that surrounds the content of the element (that is, not the box that surrounds the padding, border, or margin).
Percentage	100 and 100	Places the lower, right-hand corner of the image in the lower, right-hand corner of the element.

<b>Background Position</b>	<b>Value Pair</b>	<b>Effect</b>
Percentage	14 and 84	Places the point that is 14 percent across and 84 percent down the image at the point that is 14 percent across and 84 percent down the element.
Length	2cm and 2cm	Places the upper, left-hand corner of the image 2 centimeters (cm) to the right and 2 cm below the upper, left-hand corner of the element.
Top Left or Left Top	NA	The same as 0 percent and 0 percent.
Top Center or Center Top	NA	The same as 50 percent and 0 percent.
Right Top or Top Right	NA	The same as 100 percent and 0 percent.
Left Center or Center Left	NA	The same as 0 percent and 50 percent.
Center and Center	NA	The same as 50 percent and 50 percent.
Right Center or Center Right	NA	The same as 100 percent and 50 percent
Bottom Left or Left Bottom	NA	The same as 0 percent and 100 percent.
Bottom Center or Center Bottom	NA	The same as 50 percent and 100 percent.
Bottom Right or Right Bottom	NA	The same as 100 percent and 100 percent.

---

**Note.** If you set only one percentage or length value, the system sets the horizontal position only and the vertical position becomes the default 50 percent. If you set two values, the first one is the horizontal position. You can set negative positions and combinations of length and percentage values, for example 50 percent and 2 cm.

---

## Referencing a Background Image URL

To specify a background image URL, select from a drop-down list box that is populated by entries in the URL Maintenance table. To reference a background image file on a designated server, add its location to the URL Maintenance table.

You reference a background image in the URL Maintenance table in the same way you would register any other URL.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," URL Maintenance.

## Specifying Border Attributes

Access the class dialog box and select the Border tab.

**PSGROUPBOX (Class)**

Font | Spacing/Alignment | Background | **Border** | Margins | Miscellaneous

**Border Width**

Top:	Length	1	Pixels
Bottom:	Length	1	Pixels
Left:	Length	1	Pixels
Right:	Length	1	Pixels

**Border Color**

Top:	#A3BED8	...
Bottom:	#A3BED8	...
Left:	#A3BED8	...
Right:	#A3BED8	...

**Border Style**

Top:	Solid
Bottom:	Solid
Left:	Solid
Right:	Solid

OK Cancel

PSGROUPBOX (Class) dialog box: Border tab

**Border Width**

Specify the width of each line in the border. Values are: *Thin*, *Medium*, or *Thick*, or you can specify *Length*, which enables you to specify a font size (with a number) and the measurement type from the drop-down list box (*Pixels*, *Inches*, *Millimeters*, and so on).

**Border Color**

Specify the color of each line in the border. You can specify a hexadecimal value for the color or select the color from a color palette.

**Border Style**

Specify the style of line in the border. Values are:

*None*: No border is drawn (regardless of the <border-width> value).

*Dotted*: Border is a dotted line that is drawn on top of the background of the element.

*Dashed*: Border is a dashed line that is drawn on top of the background of the element.

*Solid*: Border is a solid line.

*Double*: Border is a double line that is drawn on top of the background of the element. The sum of the two single lines and the space between equals the <border-width> value.

*Groove*: Border is a three-dimensional groove that is drawn in colors based on the <color> value.

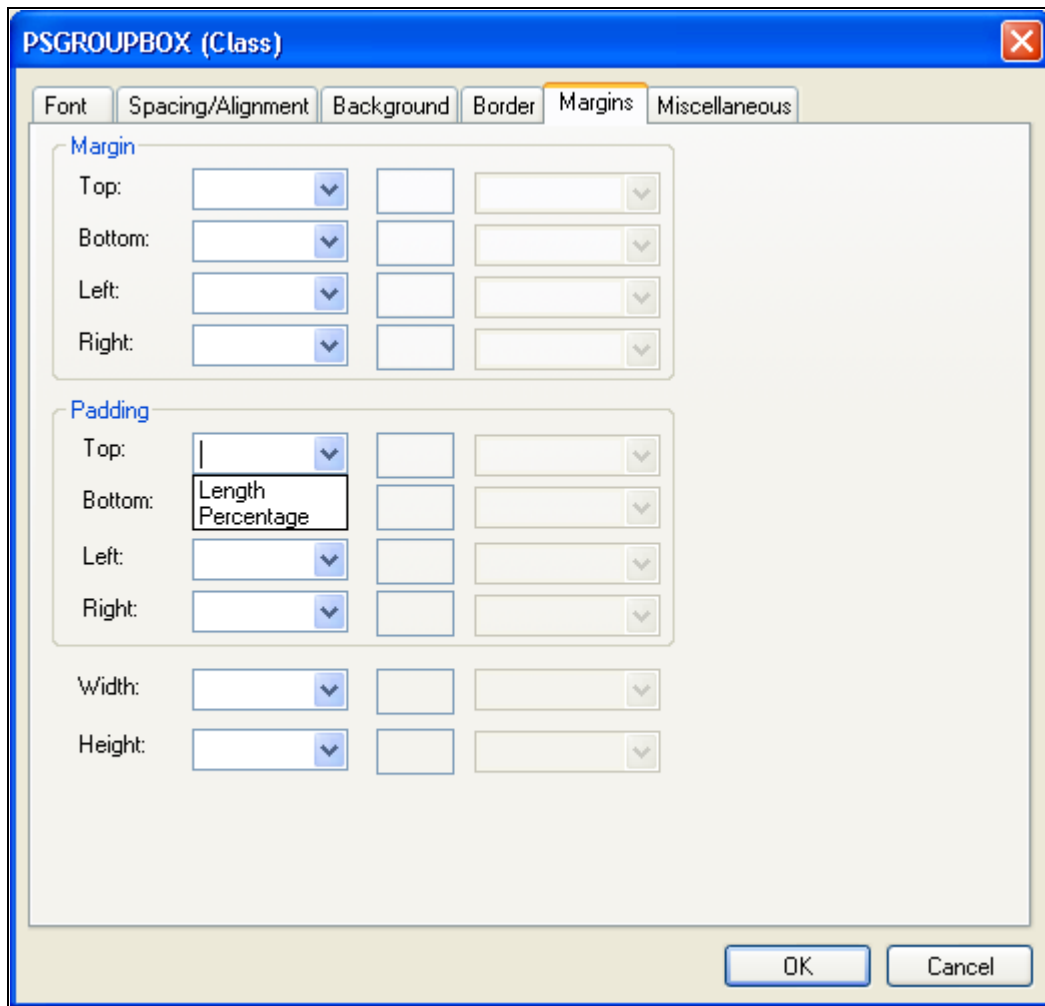
*Ridge*: Border is a three-dimensional ridge that is drawn in colors based on the <color> value.

*Inset*: Border is a three-dimensional inset that is drawn in colors based on the <color> value.

*Outset*: Border is a three-dimensional outset is that is drawn in colors based on the <color> value.

## Specifying Margins

Access the class dialog box and select the Margins tab.



PSGROUPBOX (Class) dialog box: Margins tab

### Margin and Padding

Specify the margin and padding for each side of the element. Values are:

*Length:* Specify a font size and the measurement type from the drop-down list box (*pixels, inches, millimeters*, and so on).

*Percentage:* Specify a number, followed by a percent sign (%). The percentage value is relative to the parent element.

*Auto:* The system calculates the width.

Do not use negative values for padding.

### Width and Height

Set the width and height of text elements. These properties are most useful with replaced elements, such as images. The system scales the image to fit the value that you specify. If you set either property to *Auto* and the image needs scaling, the system preserves the aspect ratio of the image. Do not use negative values.

---

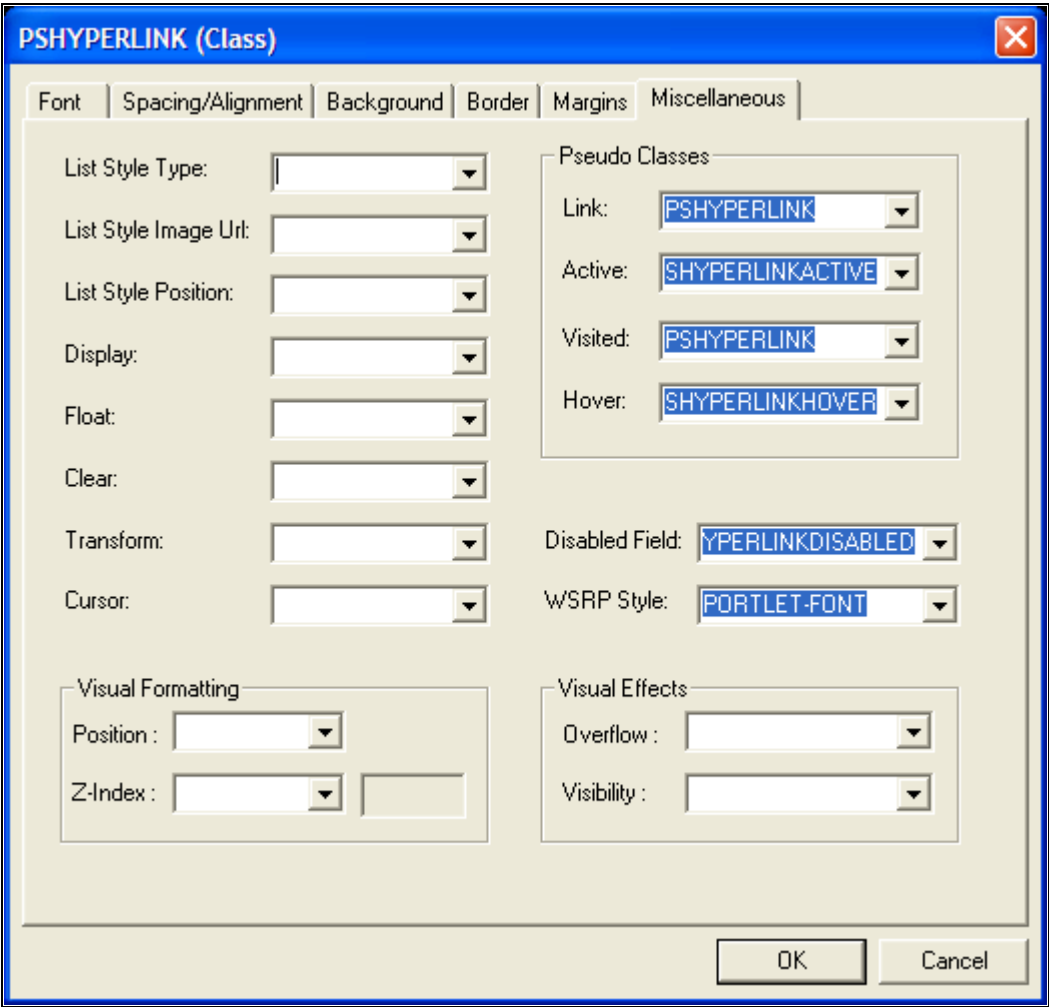
**Note.** If the width and height of a replaced element are both set to *Auto*, the system sets these properties to the intrinsic dimensions of the element.

---



# Specifying Miscellaneous Attributes

Access the class dialog box and select the Miscellaneous tab.



PSHYPERLINK (Class) dialog box: Miscellaneous tab

- List Style Type**

Specify the appearance of the list-item marker if the List Style Image URL field is blank or if the system cannot display the image that the URL referenced.
- List Style Image Url** (list style image URL)

Specify the URL containing the image to display as the list-item marker.
- List Style Position**

Specify how the list-item marker is drawn relative to the content. Values are: *Inside* and *Outside*.

**Display**

Specify how an element appears on the canvas, which might be a printed page or a computer display. Values are:

*Block*: Opens a new box. The box is positioned relative to adjacent boxes. Typically, elements like H1 and P are block types.

*ListItem*: Similar to block except that a ListItem marker is added. In HTML, LI typically has this value.

*Inline*: Results in a new inline box on the same line as the previous content. The box dimensions are based on the formatted size of the content. If the content is text, it might span several lines and each line will have a box. The margin, border, and padding properties apply to inline elements but do not affect line breaks.

*None*: Deactivates the display of the element, including children elements and the surrounding box.

**Float**

Specify how the element floats with the text. Values are:

*None*: Causes the element to appear where it appears in the text.

*Left* and *Right*: Causes the element to move to the left or right, and the text wraps on the right-hand or left-hand side of the element, respectively. With a *Left* or *Right* value, the system treats the element as block-level and ignores the Display property.

**Clear**

Specify the sides of an element on which floating elements are not accepted. Values are:

*Left*: An element appears below any floating element on the left-hand side.

*Right*: An element appears below any floating element on the right-hand side.

*Both*: An element appears below any floating element on the left-hand and the right-hand side.

*None*: Floating elements are allowed on all sides.

**Transform**

Use for text elements only. Values are: *Capitalize*, *Uppercase*, *Lowercase*, and *None*.

**Cursor**

Specify the appearance of the cursor when it passes over an element.

**Disabled Field**

Specify the appearance of the link when the link (or field) is disabled.

**WSRP Style** (web services for remote portlets style)

Specify the appearance of the link when the link appears in a portlet.

**Pseudo Classes**

Pseudo classes are mechanisms that extend the expressiveness of style sheets. Using pseudo classes, you can change the style of page links based on whether and when the links have been visited. Pseudo classes do not exist in HTML; that is, they are not visible in the HTML code.

PeopleSoft applications uses pseudo classes to specify how a browser indicates to a user the status of links in a document that the user is viewing. For example, a browser might display a document link in a different color than the rest of the text.

<b>Link</b>	Specify the normal appearance of a link.
<b>Active</b>	Specify the appearance of a link that is actively selected.
<b>Visited</b>	Specify the appearance of a link that has been visited.
<b>Hover</b>	Specify the appearance of a link when it is designated but not activated. For example, when the cursor hovers over a box that an element generated.

## Visual Formatting

<b>Position</b>	<p>The position property places an element in a static, relative, absolute, or fixed position.</p> <p><i>Static:</i> The page element always has the position the normal flow of the page gives it (a static element ignores any top, bottom, left, or right declarations). This value is the default.</p> <p><i>Relative:</i> The page element moves relative to its normal position, for example, <i>left:20</i> adds 20 pixels to the left position of the element.</p> <p><i>Absolute:</i> The page element is positioned at the specified coordinates relative to its containing block. The position of the element is specified with the <i>left</i>, <i>top</i>, <i>right</i>, and <i>bottom</i> properties.</p> <p><i>Fixed:</i> The page element is positioned at the specified coordinates relative to the browser window.</p> <p><i>Absolute:</i> The page element is positioned at the specified coordinates relative to its containing block. The position of the element is specified with the <i>left</i>, <i>top</i>, <i>right</i>, and <i>bottom</i> properties.</p>
-----------------	---

<b>Z-Index</b>	<p>Positioning of elements in CSS occurs in three dimensions, not just two. The placement of elements along this third axis, a z-axis, is exactly what this property controls. The third dimension is perpendicular to the screen, giving the screen a sense of depth. Elements can be overlapped, with higher or closer elements obscuring elements that are lower or farther away (element transparency becomes an important issue with this capability).</p> <p>The rendering box for each element is automatically assigned an integer z-index stacking level based on its context in the document. Boxes with greater z-axis numbers will appear in front of boxes with lower z-axis numbers (0 and negative numbers are allowed). Explicitly setting the z-index property for an element box not only sets its z-position relative to other element boxes in its current context, it also initiates a new stacking context hierarchy that affects the current element box and its child elements. If two or more boxes have the same stacking level within the same context, then they are rendered back to front in the order of the document tree. If no z-index property is set for an element box, then it inherits the stacking level of its parent element box.</p>
----------------	---

## Visual Effects

### Overflow

This property specifies whether the content of a block-level element is clipped when it overflows the element box (which acts as a containing block for the content).

*Visible:* This value indicates that content is not clipped; it may be rendered outside the block box.

*Hidden:* This value indicates that the content is clipped and that no scrolling mechanism should be provided to view the content outside the clipping region; users will not have access to clipped content. You specify the size and shape of the clipping region using the Clip property.

*Scroll:* This value indicates that the content is clipped and that if the user agent uses scrolling mechanism that is visible on the screen (such as a scroll bar), a box should display the mechanism whether any of its content is clipped. This setting avoids any problem with scroll bars appearing and disappearing in a dynamic environment. When this value is specified and the target medium is print or projection, overflowing content should be printed.

*Auto:* The behavior of this value is user agent-dependent, but it should cause a scrolling mechanism to be provided for overflowing boxes.

### Visibility

This property specifies whether the boxes generated by an element are rendered. Invisible boxes still affect layout (set the Display property to None to suppress box generation altogether).

*Visible:* The generated box is visible.

*Hidden:* The generated box is invisible (fully transparent) but still affects layout.

*Collapse:* This property may be used in conjunction with scripts to create dynamic effects on rows and columns. If used on elements other than rows or columns, *Collapse* has the same meaning as *Hidden*.

---

## Working With Free Form Sub Style Sheets

This section discusses how to:

- Create free form sub style sheets.
- Insert free form sub style sheets into standard style sheets.
- Convert standard style sheets into free form sub style sheets.
- Rename free form sub style sheets.
- Delete free form sub style sheets.
- Include meta variables and comments in free form sub style sheets.

- Change font and color settings in free form sub style sheets.

## Creating Free Form Sub Style Sheets

To create a free form sub style sheet:

1. Select File, New.
2. Select Style Sheet from the list.
3. Select the Free Form Sub Style Sheet style sheet type.
4. Select File, Definition Properties to enter a description for documentation purposes; when done, click the OK button.
5. Select the Free Form tab and enter the text for the style sheet.
6. Save the style sheet.

---

**Important!** The Style Sheet Editor does not perform any syntax checking on free form sub style sheets. You must verify and correct any style errors that occur at runtime.

---

## Inserting Free Form Sub Style Sheets into Standard Style Sheets

To insert a free form sub style sheet into a standard style sheet:

1. Open the receiving standard or sub style sheet.
2. Click the Fixed Form tab.
3. Select Insert, Insert Sub Style Sheet.
4. Select *Freeform Sub Style Sheet* from the Sub Style Sheet Type drop-down list box.
5. Select the free form style sheet from the Sub Style Sheet drop-down list box.
6. Click the OK button and save the style sheet.

## Converting Style Sheets to Free Form Sub Style Sheets

To convert a style sheet to a free form sub style sheet:

1. Open a standard style sheet or sub style sheet.
2. Select File, Definition Properties.
3. In the Style Sheet Type drop-down list box, select *Freeform Sub Style Sheet* and click the OK button.

---

**Note.** You will be warned that once the style sheet is saved as a free form type the style sheet data cannot be converted back to the standard or sub style sheet type.

---

4. Dismiss the warning dialog box.

5. Save the style sheet.

---

**Note.** You should not modify a PeopleTools style sheet; instead, clone the style sheet and make changes to the copy.

---

## Renaming Free Form Sub Style Sheets

To rename a free form sub style sheet:

1. Select File, Rename.
2. Select *Style Sheet* in the Definition drop-down list box and then enter search criteria to find the style sheet definition that you are renaming.
3. Click the Rename button; the definitions that match the search criteria appear in the list box.
4. Double-click the style sheet definition that you want to rename.
5. Enter the new name of the style sheet definition in the highlighted area and press the Enter key.
6. Click the OK button to confirm the rename.

---

**Note.** When you rename a free form sub style sheet, PeopleSoft Application Designer automatically renames all references to the style sheet.

---

## Deleting Free Form Sub Style Sheets

To delete a free form sub style sheet:

1. Select File, Delete.
2. Select *Style Sheet* in the Definition drop-down list box and then enter search criteria to find the style sheet definition that you are deleting.
3. Press the Delete key; the definitions that match the search criteria appear in the list box.
4. Click the style sheet that you want to delete and press the Delete key.
5. Click the OK button to confirm the deletion or click the Cancel button to cancel the operation.

---

**Note.** When you delete a style sheet, PeopleSoft Application Designer automatically deletes all references to the deleted style sheet.

---

## Including Meta Variables and Comments in Free Form Sub Style Sheets

This section discusses:

- Meta variables.
- Comments.

## **Meta Variables**

Free form sub style sheets support these PeopleTools meta variables:

- %AlignStart
- %AlignEnd
- %AlignAmount
- %Image

See *PeopleTools 8.52: PeopleCode Language Reference*, "Meta-HTML."

See *PeopleTools 8.52: Global Technology*, "Implementing Bidirectional Language Support," Developing HTML Areas.

---

**Note.** The system ignores any other meta variables.

---

Meta variables in the free form sub style sheet are resolved at runtime before the style sheet is downloaded to the client. Meta variable processing does not occur at design time. In PeopleSoft Application Designer, if you select Layout, View in Browser, you do not see any results from meta variable processing.

## **Comments**

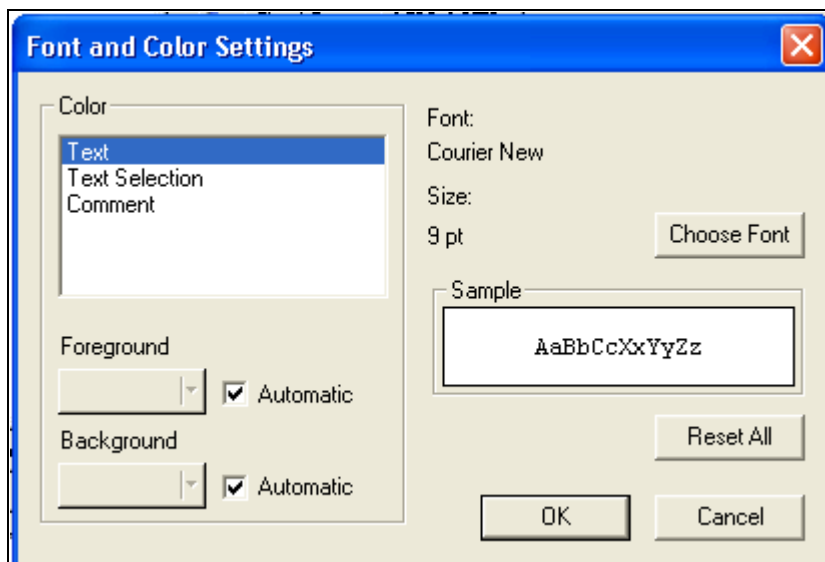
To insert comments, include the text between /\* and \*/.

The use of // is not supported.

## **Changing Font and Color Settings in Free Form Sub Style Sheets**

When entering code in the Free Form tab of free form sub style sheets, the appearance of the various elements of the code, such as comments, is based on the default font and color settings in the Font and Color Settings dialog box.

Access the Font and Color Settings dialog box. (Open a free form sub style sheet and select Edit, Display Font and Colors.)



Font and Color Settings dialog box

<b>Color</b>	Select the item whose color you want to change. Values are: <i>Text</i> <i>Text Selection</i> <i>Comment</i>
<b>Foreground</b>	Click to select a font color.
<b>Background</b>	Click to select the color of the background behind the font.
<b>Choose Font</b>	Click to select a font.
<b>Reset All</b>	Click to reset all defaults to the original settings.

You can change the default font and color settings for free form sub style sheets.

### ***Changing the Default Font and Color Settings***

To change a default font and color setting:

1. In the Color group box, select an item from these choices:  
*Text*  
*Text Selection*  
*Comment*
2. Select a color from the Foreground drop-down list box.
3. Select a color from the Background drop-down list box.
4. Click the Choose Font button to select another font.



5. Click Reset All to restore the settings to the default settings.
6. Click the OK button.

---

## Working with the Default Style Sheet

Applications use a default style sheet that contains all default style classes for PeopleSoft applications. You specify the default application style sheet on the PeopleTools Options, System Options, and Registry Options pages. When you install PeopleSoft applications, PSSTYLEDEF is the application style sheet for PeopleSoft applications before and including application release 9.0. PSSTYLEDEF\_SWAN is the default application style sheet when you install PeopleSoft applications for releases 9.1 and newer.

This section discusses:

- Configuring PSSTYLEDEF as the default style sheet.
- Configuring PSSTYLEDEF\_SWAN as the default style sheet.
- Configuring other default style sheets

---

**Note.** You can only use standard style sheets as a default application style sheet.

---

**Important!** Oracle has updated the styles that define the PeopleSoft user interface. This PeopleSoft PeopleTools release delivers the classic (old) style as well as two new styles: a dark blue style and a light blue style. PeopleSoft 8.4 applications and pre-8.50 PeopleSoft PeopleTools system databases use the classic style, but all other applications use the new dark blue style. The dark blue style is set as the default. To use another interface, you have to delete the sub style sheets associated with the dark blue style and replace them with either the classic or light blue sub style sheet. The classic and light blue styles are considered deprecated as of PeopleTools 8.50.

To enable a deprecated user interface, consult your upgrade and installation documentation.

---

**Important!** After installing PeopleSoft translations using the Global Multi-Language CD-ROM, you must rerun the UPGPT850SWAN Application Engine program if your application uses a SWAN style sheet as the default style sheet.

---

## Configuring PSSTYLEDEF as the Default Style Sheet

PSSTYLEDEF is the default application style sheet. The PSSTYLEDEF style sheet comprises all default style classes and consists of these sub style sheets:

- JSR168STYLEDEF: Defines PeopleSoft Portlet default style classes.
- PAALTERNATE: Defines PeopleSoft Portal Solutions alternatives to PS default style classes.
- PSACE: Defines analytic calculation engine and analytic grid default style classes.
- PSALTERNATE: Defines application alternatives to PS default style classes.
- PSCHARTSTYLE: Defines charting default style classes.

- **PSORGCHART**: Defines organization chart default style classes.
- **PSRATEBOX**: Defines rating box default style classes.
- **PTSTYLEDEF**: Defines the standard PeopleSoft Pure Internet Architecture and portal default style classes.
- **PSNAV2**: Defines the 8.5x Menu navigation style classes.

This sub style sheet is free form.

- **PSPOPUP\_CSS**: Defines the pop-up dialog box page style classes.

This sub style sheet is free form.

- **PSTAB\_PTCSS**: Defines the page tab style classes.

This sub style sheet is free form.

---

**Note.** You should not modify **PSSTYLEDEF** or other PeopleTools style sheets; instead, clone existing style sheets and make changes to the copy.

---

To change the default application style sheet to **PSSTYLEDEF**:

---

**Note.** **PSSTYLEDEF** is the default; you should only perform these steps if you are reverting to **PSSTYLEDEF** after implementing another style sheet.

---

1. In the browser, select PeopleTools, Utilities, Administration, PeopleTools Options.
2. Enter *PSSTYLEDEF* in the Style Sheet Name field.
3. Save the page.
4. Select PeopleTools, Portal, Portal Utilities, System Options.
5. Enter *PSSTYLEDEF* in the Style Sheet Name field.
6. Save the page.
7. In the browser, select PeopleTools, Portal, Portal Utilities, Registry Options.
8. Select the desired registry.
9. Enter *PSSTYLEDEF* in the Style Sheet Name field.
10. Save the page.
11. Run the UPGPT850SWAN Application Engine program.
12. Stop the web server, delete all cache files, and restart the web server.
13. Stop the application server, delete all cache files, and restart the web server.

## Configuring PSSTYLEDEF\_SWAN as the Default Style Sheet

The style classes in the PSSTYLEDEF\_SWAN style sheet are comparable to those in the PSSTYLEDEF style sheet. The page styles provided by PSSTYLEDEF\_SWAN are the standard design for all Oracle Applications Unlimited products.

---

**Note.** The My Reports pagelet contains exceptions to the SWAN style sheet and might appear to look different from the standard SWAN style.

---



---

**Note.** Any custom style sheets or sub style sheets that you use with PSSTYLEDEF\_SWAN must include SWAN in the definition name.

---

- JSR168STYLEDEF\_SWAN: Defines PeopleSoft Portlet default style classes.
- PAALTERNATE\_SWAN: Defines PeopleSoft Portal Solutions alternatives to PS default style classes.
- PSACE\_SWAN: Defines analytic calculation engine and analytic grid default style classes.
- PSALTERNATE\_SWAN: Defines application alternatives to PS default style classes.
- PSCHARTSTYLE\_SWAN: Defines charting default style classes.
- PSORGCHART\_SWAN: Defines organization chart default style classes.
- PSRATEBOX\_SWAN: Defines rating box default style classes.
- PTSTYLEDEF\_SWAN: Defines the standard PeopleSoft Pure Internet Architecture and portal default style classes.
- PSNAV2\_SWAN: Defines the 8.5x Menu navigation style classes.

This sub style sheet is free form.

- PSPOPUP\_CSS\_SWAN: Defines the pop-up dialog box page style classes.

This sub style sheet is free form.

- PSTAB\_PTCSS\_SWAN: Defines the page tab style classes.

This sub style sheet is free form.

---

**Note.** You should not modify PSSTYLEDEF\_SWAN or other PeopleTools style sheets; instead, clone existing style sheets and make changes to the copy.

---

To change the default application style sheet to PSSTYLEDEF\_SWAN:

1. In the browser, select PeopleTools, Utilities, Administration, PeopleTools Options.
2. Enter *PSSTYLEDEF\_SWAN* in the Style Sheet Name field.
3. Save the page.
4. Select PeopleTools, Portal, Portal Utilities, System Options.
5. Enter *PSSTYLEDEF\_SWAN* in the Style Sheet Name field.

6. Save the page.
7. In the browser, select PeopleTools, Portal, Portal Utilities, Registry Options.
8. Select a registry.
9. Enter *PSSTYLEDEF\_SWAN* in the Style Sheet Name field.
10. Save the page.

---

**Note.** Repeat Steps 7 through 10 for each portal registry.

---

11. Run the UPGPT850SWAN Application Engine program.
12. Stop the web server, delete all cache files, and restart the web server.
13. Stop the application server, delete all cache files, and restart the application server.

---

**Important!** After installing PeopleSoft translations using the Global Multi-Language CD-ROM, you must rerun the UPGPT850SWAN Application Engine program if your application uses a SWAN style sheet as the default style sheet.

---

---

**Important!** Do not use the \*\_SWAN images in any customized code. These images are used by an Application Engine program to toggle between the new appearance and classic appearance. Using these images or modifying them in any way changes the attributes and has unexpected distortions in the appearance of your application pages. In addition, these images are subject to change in future releases of the application.

---

## Configuring Other Default Style Sheets

You can use other style sheets as the default, including those that comprise free form text only, as long as the style sheet contains all default style classes.

To change the default application style sheet:

1. In the browser, select PeopleTools, Utilities, Administration, PeopleTools Options.
2. Enter the style sheet in the Style Sheet Name field.
3. Save the page.
4. Select PeopleTools, Portal, Portal Utilities, System Options.
5. Enter the style sheet name in the Style Sheet Name field.
6. Save the page.
7. In the browser, select PeopleTools, Portal, Portal Utilities, Registry Options.
8. Enter the style sheet name in the Style Sheet Name field.
9. Save the page.
10. Stop the web server, delete all cache files, and restart the web server.
11. Stop the application server, delete all cache files, and restart the application server.

---

## Overriding Default Page Style Properties

Unless you override the default style sheet at the page level, the browser renders all pages and page fields based on the default style classes that comprise the application style sheet. If you want to use different styles to create subtle changes or entirely unique pages, you can override styles at the page and page field levels.

This section discusses how to:

- Override the default page style sheet.
- Override the default page background style.
- Override default page field styles.

## Overriding the Default Page Style Sheet

You can override the default page style sheet by accessing the page properties dialog box of any PeopleSoft page. When you override the default page style sheet, you are not changing the default style classes for the page; you are indicating to the system to select the default style classes from a nondefault style sheet. Note these points when overriding the page style sheet:

- Changing the page style sheet overrides the default style sheet only for the selected page.
- Only the custom and default style classes that comprise the current page style sheet—whether default or override—are available for use by the page and its elements. Style classes on other style sheets are inaccessible.
- If you select a page style sheet that comprises only free form sub style sheets, the system constrains all subordinate page elements to their default style classes.
- If you select a page style sheet that comprises only free form sub style sheets, the page designer renders the page using the application style sheet, not the free form style sheet.

To override the default page style sheet:

1. Open the page in PeopleSoft Application Designer.
2. Select File, Definition Properties to access the Page Properties dialog box.
3. Select the Use tab.
4. Click the Page Style Sheet drop-down list box and select a style.
5. Click the OK button and save the page.
6. Select Layout, View in Browser to preview the page.

Unlike the page designer view, this option renders the page with all style classes, including free form styles.

## Overriding the Default Page Background Style

The page background derives from the Background tab of the PSPAGE style class contained in the current page style sheet, whether default or override. You can override the default page background style.

Note these points about changing the default value of the page background style:

- Changing the page background style value for a specific page overrides the background properties of only the current page.
- If the current page style sheet comprises only free form sub style sheets, the system constrains the page background style to the default style class.
- The Page Background Style drop-down list box displays only the style classes that comprise the current page style sheet, whether default or override. Style classes that do not appear in the current page style sheet do not appear in the list.

To override the default page background style:

1. Open the page in PeopleSoft Application Designer.
2. Select File, Definition Properties to access the Page Properties dialog box.
3. Select the Use tab.
4. Click the Page Background Style drop-down list box and select a style.
5. Click the OK button and save the page.
6. Select Layout, View in Browser to preview the page.

Unlike the page designer view, this option renders the page with all style classes, including free form styles.

## Overriding the Default Page Field Styles

You can override the default style of most page field controls, such as edit boxes, drop-down list boxes, and so on. Many page field controls include a separate label style that enables you to override the label style independent of the control style itself.

Note these points about changing the default style of a page field:

- The Style drop-down list box displays only the style classes that comprise the current page style sheet, whether default or override. Style classes that do not appear in the current page style sheet do not appear in the list.
- If the page style sheet comprises only free form sub style sheets, then the system constrains all page field controls to the default style despite the Style field value.

To override the default page field style:

1. Open a page in PeopleSoft Application Designer.
2. Double-click the page field control to access the Properties dialog box.

3. Select the Record tab.
4. Click the Style drop-down list box and select a new style.
5. Click the Label tab.
6. Click the Style drop-down list box and select a new style.
7. Select Layout, View in Browser to preview the page.

Unlike the page designer view, this option renders the page with all style classes, including free form styles.

8. Save the page.

In addition to setting the style property at design time, you can also use the PeopleCode Field object Style property to programmatically change the style class of a field at runtime.

See *PeopleTools 8.52: PeopleCode Developer's Guide*, "PeopleCode and PeopleSoft Pure Internet Architecture," Using the Field Object Style Property.

---

## Modifying Page Tabs

This section provides an overview of tabs and discusses how to:

- Create tab images.
- Customize page tab style sheets.

## Understanding Tabs

To provide a greater degree of and simpler user-level configuration, PeopleSoft applications use the sliding door technique for tabs. Cascading style sheets (CSS) have the advantage of layering background images, which enables them to slide over each other to create visually interesting effects.

The default style sheet and the SWAN style sheet contain CSS tabs. You can, however, change the background colors of these CSS folder and grid tabs. To change the background colors of tabs, you should have a basic understanding of how tab definitions are constructed and stored in the database.

### ***Sliding Doors Technique***

Sliding door tabs expand and contract based on the size of the text by using two separate background images: one image to render the left side of a tab and the other image to render the right. Think of these two images as the sliding doors that complete one opening. The doors slide together and overlap more to fill a narrower space, or slide apart and overlap less to fill a wider space. The background images only show in the available doorway of the element to which they are applied (content area plus padding). The two images are anchored to the outside corners of their respective elements. The visible portions of these background images fit together inside the doorway to form a tab-like shape.

## ***Sliding Door Tab Images***

You need to know the following image names to modify the style sheets that ship with PeopleSoft applications.

For the dark blue look and feel, folder tabs and grid tabs use these images:

- PSTAB\_LEFT.gif
- PSTAB\_RIGHT.gif

For the SWAN look and feel, the folder tabs and the grid tabs use these images:

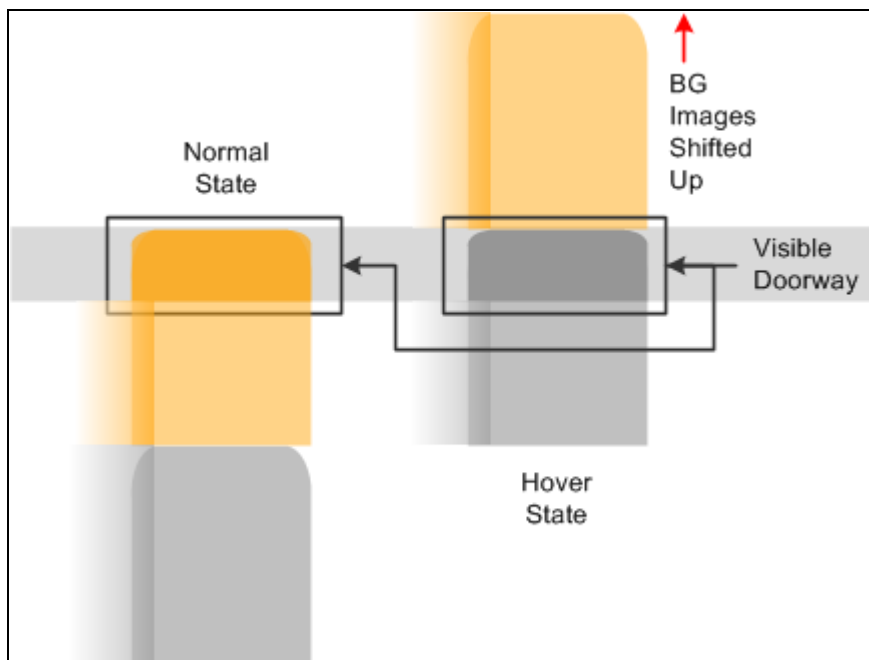
- PSTAB\_LEFT\_SWAN.gif
- PSTAB\_RIGHT\_SWAN.gif

The CSS code behind the rendering of the tab image is:

```
#PSTAB a
{
background:transparent url(%IMAGE(PSTAB_LEFT)) no-repeat left top;
padding:0 2px 0 5px;
}
#PSTAB a span
{
font-size:9pt;
background:transparent url(%IMAGE(PSTAB_RIGHT)) no-repeat right top;
padding:4px 14px 2px 5px;
}
```

When working with sliding door tabs, the images load faster, are simpler to maintain, and the text within them can be scaled up or down significantly in size without breaking the page. Along with these new images, applications take advantage of the CSS background-position property to move into view the appropriate portion of the combined-state background image when the user hovers over a tab.





Style sheet sliding door tab preload technique

Applications use the shifted background positions from the active, inactive, and hover states.

## Creating Tab Images

Before you can assign a new background color to the active, inactive, and hover states of the tabs, you must create the necessary tab images in that color. To produce tabs in a different color combination:

1. In PeopleSoft Application Designer, open the image definitions for the tabs that you want to change.
2. Select File, Export Image and, in the Save Image dialog box, select the directory in which you want to place the images.
3. Use any image editor to alter the image colors to the same as the new background color you choose.

---

**Note.** Do not replace any of the transparent pixels with solid pixels. Additionally, you should leave the size and shape as they are even though the style sheets allows you to alter the size and shape of images.

---

4. Select File, New, Image to add the new image definitions to the database.
5. Save the new image files under the appropriate names.
6. Customize the tabs.

See [Chapter 15, "Creating Style Sheet Definitions," Customizing Page Tab Style Sheets, page 484.](#)

## Customizing Page Tab Style Sheets

After you create the new images, you must change the properties of the style classes rendering the new sliding door tabs. The style classes that render the folder tabs and the grid tabs are PSTAB and PTGRIDTAB, respectively. Both style classes are part of the PSTAB\_PTCSS free form style sheet.

To modify the tab properties:

1. Open the appropriate free form style sheet, PSTAB\_PTCSS if you are using the default style sheet or PSTAB\_PTCSS\_SWAN if you are using the SWAN style sheet.
2. For folder tabs, modify the PSTAB style class; for grid tabs, modify the PTGRIDTAB style class.

For example, if you have two new folder tab images, the left and the right image of the tab, each has a different name and you must change the code as you see here:

```
#PSTAB a
{
background:transparent url(%IMAGE(<newlyCreatedLeftSideTabImageName>)) no-repeat=
  left top;
....
}
#PSTAB a span
{
font-size:9pt;
background:transparent url(%IMAGE(<newlyCreatedRightSideTabImageName>)) no-repeat=
  right top;
....
}
```

---

**Note.** You must replace <newlyCreatedLeftSideTabImageName> and <newlyCreatedRightSideTabImageName> with the names of the new images that you create. You might have to modify the position property if the new images vary in size from the original images.

---

3. Because the image contains all active, inactive, and hover color properties, you only need to change the pixel properties and background positioning if the delivered images have changed in size and shape.
4. You can modify style class properties such as border, padding, font, and text to create the required look for the tabs you are implementing.
5. You can override the tab bottom line by changing the following code:

```
#PSTAB #TABBOTTOMLINE
{
Color: #809EBA;
background: repeat-x right 18px;
}
```

6. You can adjust the hover properties as shown in this example:

```
#PSTAB a:hover a
{
color: #555555;
}

#PSTAB a:hover span
{font-size:9pt;
background-position:100% -50px;
}
```

---

**Note.** The `<span>` element renders the main part of the tab. Adjust any necessary properties, such as `background-position` in the `<span>` element to render the image correctly. The left image is small, and you need to adjust only the color and possibly the border properties.

---

7. Save the style sheet.
8. Stop the application server, clear the cache, and restart the application server.
9. Stop the web server, clear the cache, and restart the web server.

---

## Modifying Push Buttons in SWAN Style Sheets

This section provides an overview of push buttons in the SWAN style sheet and discusses how to:

- Create SWAN style sheet push button images.
- Customize SWAN push buttons in HTML areas.

## Understanding Push Button in the SWAN Style Sheet

PeopleSoft applications provide the PTPUSHBUTTON free form style sheet as part of the SWAN style sheets. You can edit all properties in the PTPUSHBUTTON style sheet to change the look and feel of the classic yellow push button. Like sliding door tabs, push buttons expand and collapse to accommodate the button text, and the buttons come with a left and right image. The left image is associated with the `<a>` element, and the right image is associated with the `<span>` element. The `<span>` element contains text and font properties for the button. The PTPUSHBUTTON free form style sheet overrides the existing properties provided in the PSPUSHBUTTON style class in the PSSTYLEDEF\_SWAN style sheet. You edit push buttons similarly to the way that you edit sliding door tabs.

### See Also

[Chapter 15, "Creating Style Sheet Definitions," Sliding Door Tab Images, page 482](#)

## Creating SWAN Style Sheet Push Button Images

All the steps for altering sliding door tab images are the same as for sliding door buttons. Do not alter the size of a button because doing so will affect the way the background position is defined for the button.

**See Also**

Chapter 15, "Creating Style Sheet Definitions," Sliding Door Tab Images, page 482

**Customizing SWAN Push Buttons in HTML Areas**

If any custom-defined push buttons in HTML areas appear in the yellow button background, you can change them to look like the SWAN push button style. To customize SWAN push buttons in HTML areas:

1. Open the HTML area that contains the push button you want to modify.
2. Locate the `<input type=button.../>` tag.
3. Following the `<input type...>`tag, insert the following tags: `<a class='PSPUSHBUTTON'><span> ... </span></a>`.

## Chapter 16

# Creating Image Definitions

This chapter provides an overview of image definitions and discusses how to:

- Create new image definitions.
- Open image definitions.
- Update image definitions.
- Convert image definitions to JPG format.
- Consolidate images.

---

## Understanding Image Definitions

Image definitions are image files that you import, store, and manage using PeopleSoft Application Designer. Image definitions, similar to field definitions, are reusable. After you create image definitions, you can associate them with page controls and reference them in code as often as you want. Most images that you see in PeopleSoft applications are image definitions. Image definitions appear throughout PeopleSoft applications, and serve many purposes, such as:

- In the page headers as a logo to enhance the appearance of the application.
- In Pagelet Wizard as numbers to sequence the individual tasks in a process.
- In the Main Menu as folders to indicate subordinate menus.
- In scroll area and grid headers to navigate among rows of data.
- In charts as connectors to indicate relationships among nodes.

PeopleSoft Application Designer enables you to import all types of graphics or image files and store them in the image catalog. After you finish this process, you can access the image definition and use it throughout the PeopleSoft application.

---

**Note.** PeopleTools includes many image-related features. Image *definitions* are discussed in this chapter; the other image-related features, such as fields and page controls, are discussed separately in this PeopleBook.

---

See [Chapter 4, "Creating Field Definitions," Specifying Image Field Attributes, page 73](#); [Chapter 4, "Creating Field Definitions," Specifying ImageReference Field Attributes, page 73](#); [Chapter 9, "Using Page Controls," Setting Static Image Properties, page 219](#) and [Chapter 9, "Using Page Controls," Setting Image Properties, page 220](#).

## ***Image Catalog***

PeopleSoft Application Designer stores all image definitions in the PSCONTDEFN PeopleTools table. The collective set of image definitions that exist in this table is the image catalog. The image catalog holds all PeopleTools image definitions in addition to any image definitions that you create.

## ***Image File Types***

Image file types or formats are a systematized means of storing digital images. File types vary in purpose and quality based on factors such as compression and color palette. You might want to use different file types for different images because of business or technical considerations that apply to your PeopleSoft implementation or because quality is more important to you than file size. The three most common file types that developers use most often to display images on the Internet are Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG) (JPG), and Portable Network Graphics (PNG).

Some developers believe, for example, that JPG is the best format for photographic images because it provides the greatest compression of any bitmap format in common use. However, some believe that the JPG file type is not as effective in compressing text and drawings as it is at compressing photographs. Similarly, many designers consider the GIF file type the optimal format for line drawings and text because it limits the color palette and compresses text in a way that retains the sharpness of images such as logos. The GIF file type also supports features such as image transparency and animation. These same characteristics of limited color and text compression make this file type less than optimal as the format for photographic images. Due to the proprietary nature of the GIF format, the PNG file type evolved as the free open-source response to the patented GIF format. The robust PNG file type supports more colors than the GIF, offers excellent online viewing, and streams well on modern displays.

While PeopleSoft Application Designer can create image definitions from image files that use the wireless bitmap (WBMP) format, these image definitions are not visible in the runtime environment. You must view them through a mobile device at runtime. Because WBMP image types do not appear in the design time environment, PeopleSoft Application Designer offers you the option to provide an alternate image type of the image, such as JPG, for viewing purposes only. The image that you view at runtime in the wireless application is still the original WBMP image.

See [Chapter 16, "Creating Image Definitions," Importing and Creating Alternate Image Types, page 491](#).

You should make your choice of file types by weighing the benefits and detractors of the image file types in the context of your implementation and technical requirements. PeopleSoft Application Designer can process image files regardless of file type, convert them into image definitions, and maintain them in the image catalog.

## ***Image File Size***

The maximum image file size depends on the database platform that you are using. Some database platforms support much larger image sizes, while others limit the size. PeopleSoft Application Designer displays a warning message if the image size is greater than 32 kilobytes.

## **Understanding the Image Catalog**

PeopleSoft Application Designer stores all image definitions in the PSCONTDEFN PeopleTools table. The collective set of image definitions that exist in this table is the image catalog. The image catalog holds all PeopleTools image definitions in addition to any image definition that you create. The image catalog stores all image definitions and makes them available for use by developers.

## ***PeopleTools Image Definitions***

In general:

- Most PeopleTools image definitions start with *PT\_*.
- Any image definition with *\_D* at the end of the name is the disabled mode of the image.
- All images that start with *PT\_TAB* are used for tab construction and colors.

A list of the PeopleTools images most common to all PeopleSoft application is located in an appendix in this PeopleBook.

---

## **Creating Image Definitions**

This section discusses how to:

- Create image definitions.
- Set image properties.
- Import and create alternate image types.

## **Creating Image Definitions**

To create an image definition from an image file:

1. Select File, New.
2. Select Image.
3. Click OK to access the Open Image File dialog box.
4. Select an image file.

---

**Note.** PeopleSoft applications support animated GIF files. You will not see the animation in PeopleSoft Application Designer at design time; however, you will see the animation on the page at runtime.

---

---

**Note.** Some browsers do not support all image types.

---

See *Troubleshooting Browser Limitations* available on the My Oracle Support website.  
[support.oracle.com](http://support.oracle.com).

5. Select an image file name.

PeopleSoft Application Designer displays a warning message if the image size is greater than 32 kilobytes. The maximum image size depends on the database platform that you are using. Some database platforms support much larger image sizes, while others limit the size.

6. Select Open.

The image appears as an image definition in the definition workspace.

7. Select File, Save to save the image definition in the PeopleSoft image catalog.

The Image Properties dialog box appears.

## Setting Image Properties

After you create a new image definition, you can set the image properties.

To set image properties:

1. Access the Image Properties dialog box.

If it is not already open, select File, Definition Properties.

2. (Optional) Enter a description on the General tab.
3. Select the Use tab to view image use properties.

### Image Format

Specify the main and alternate image formats. The main image format, which appears at runtime, is the format in which the image was imported, such as BMP. The alternate format is the optional image format that you can specify if you want a WBMP image as the main image format that appears at runtime.

### Image URL

Specify a reference to an image on a web server. This reference can be used as an alternative to storing an image definition in the database after an image definition is saved. When a user opens a page containing a control that is associated with this image definition, the application retrieves the image from the URL rather than from the database. Set up URLs in the URL Maintenance utility.

See *PeopleTools 8.52: System and Server Administration*, "Using PeopleTools Utilities," URL Maintenance.

### Mirror this image in right-to-left contexts

Select this property to have the system flip the image horizontally when a language that is read right-to-left (RTL) is in use.

Pages in languages with an RTL reading order, for example, Arabic, might also display images whose directionality depends on the directionality of the language. For example, an arrow in a margin that points to the text in a left-to-right (LTR) language, such as English, will point to the right; but in an RTL language, the arrow should point to the left.

### Used in Workflow Maps

Select to filter images that appear in a list box for the Change Icon function for PeopleSoft Workflow designers.



## Importing and Creating Alternate Image Types

To import a WBMP image type:

1. Select File, New.
2. Select Image.

The Open Image File dialog box appears.

3. Select *WBMP* from the Files of Type drop-down list box.
4. Locate the image that you want to import.
5. Click the Open button.

The Content Repository Interface dialog box appears, prompting you to select an alternate image.

6. Select Yes.

The Open Alternate Image File dialog box appears, prompting you to select an alternate image type. You must have the same image stored in the directory under a different file type, such as JPG, to view the image in PeopleSoft Application Designer. If you select No, you can still open the WBMP image definition, but you cannot see it. Instead, you receive a message that the image cannot be viewed in PeopleSoft Application Designer.

7. Select the alternate image format and the appropriate file.
8. Click Open.
9. Select File, Save As.

Upon saving, the Image Properties dialog box appears. If you select the Use tab, the Image Format group box shows the main image format as WBMP and the alternate image format as JPG. Now you can view the image in PeopleSoft Application Designer as a JPG image and still display the image at runtime as WBMP.

---

## Opening Image Definitions

To open an image definition:

1. Select File, Open.
2. Select *Image* from the Definition drop-down list box.
3. Click Open to view the full list of image definitions and brief descriptions.

---

## Updating Image Definitions

This section discusses how to:

- Update image definitions.
- Change an image display size.
- Specify an image storage format.

## Updating Image Definitions

Occasionally, the original image file from which you created an image definition might change. To keep the image definitions current, you might want to update them rather than create entirely new image definitions.

To update an image definition:

1. To change the image for a saved image definition, right-click the open definition and select Update Image.

The Open Image File dialog box appears, in which you can select the changed image file to replace the open image definition.

2. Click Open.

The previous image in the image definition is replaced with the new image that you selected.

3. Select File, Save to save the current image definition with the new image.

## Changing an Image Display Size

If an image file is smaller than you want it to appear in the open image definition, you can increase the size by zooming in on the definition workspace. The size of the image in the image catalog does not change.

To change an image definition display size:

1. Open the image.
2. Right-click the image and select Zoom to see the different size ratios.
3. Select a new image display percentage.

The image automatically changes to that selection.

## Specifying an Image Storage Format

You can specify a default format in which image definitions are stored. For example, if JPG is specified as the preferred storage format, then a BMP image is stored as a JPG after it is converted to an image definition.

To specify an image definition storage format:

1. Select Tools, Options to open the Options dialog box.

2. Select the Image tab.

**No conversion**

Select to import all image definitions in their original formats. This value is the default.

**DIB and JPG**

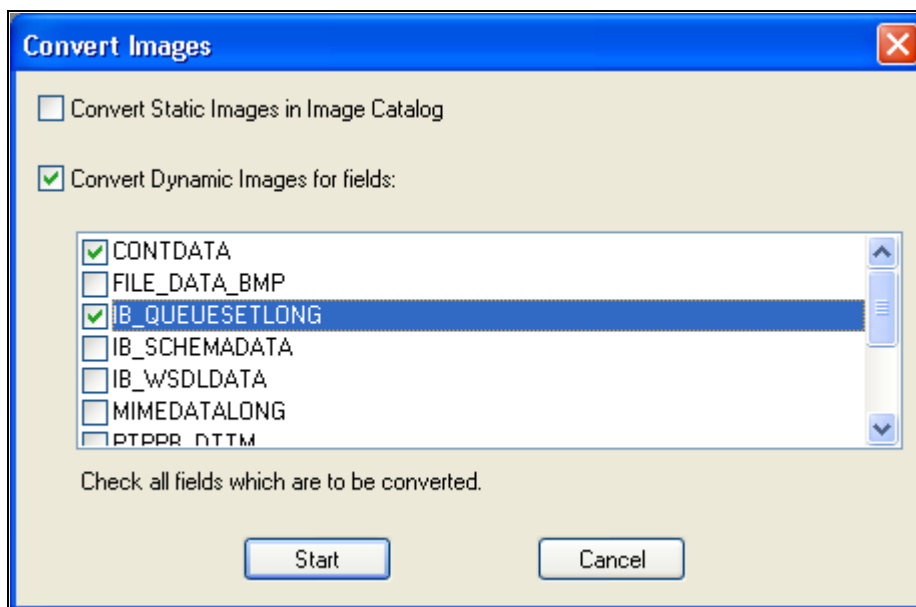
Select to convert and store the imported image definitions as either DIB or JPG in the PeopleSoft Application Designer image catalog. GIF and WBMP images cannot be converted to DIB or JPG.

3. Click the OK button.

---

## Converting Images to JPGs

Not all browsers support all image formats, but most browsers support the JPG image type. Therefore, PeopleSoft Application Designer has a utility to convert all application images to JPG.



Convert Images dialog box

To convert images to JPG format:

1. Select Tools, Upgrade, Convert Images.
2. Select one of these check boxes:

**Convert Static Images in Image Catalog**

Converts all image definitions that are stored in the image catalog of the PeopleSoft database.

**Convert Dynamic Images for fields**

Converts all images that are dynamically referenced by PeopleCode to appear in image fields.

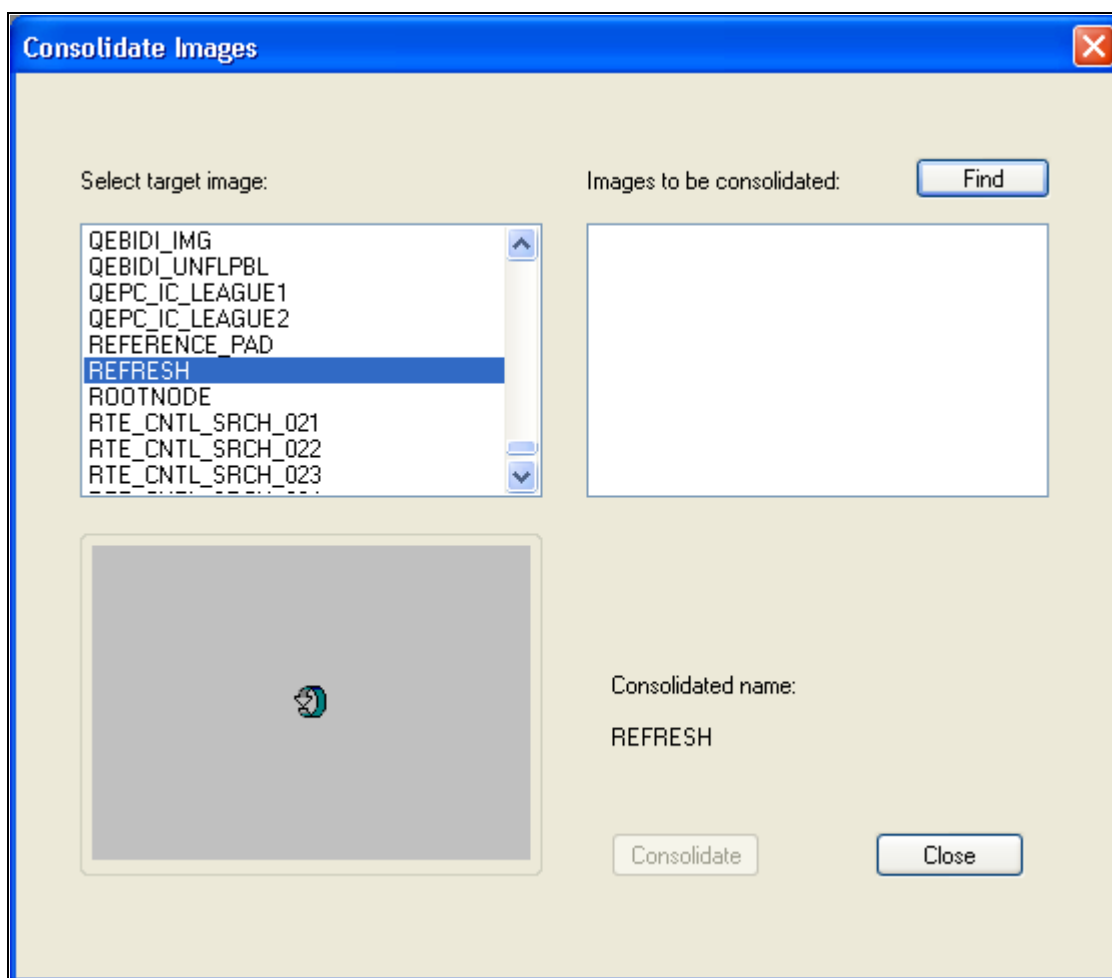
3. Select the fields to convert.
4. Click the Start button.

When the process is complete, a confirmation message appears in the Image tab of the output window.

---

## Consolidating Images

Use the Consolidate Images tool if you create custom image definitions and use them in multiple record definitions. Consolidating images helps you avoid having the same image stored in several places. All image definitions included in the PeopleSoft system have already been consolidated.



Consolidate Images dialog box

To consolidate custom image definitions:

1. Open an existing image definition in the definition workspace.
2. Select Tools, Consolidate Images.
3. Select a target image from the list.

4. Click Find to gather all of the consolidated image candidates.
5. Select all of the images.
6. Click Consolidate.

The selected image definitions are removed from the Images to be consolidated list box and consolidated into the target image.

7. Click Close.



## Chapter 17

# Creating HTML Definitions

This chapter provides an overview of HTML definitions and discusses how to:

- Create HTML definitions.
- Open HTML definitions.
- Reference HTML definitions dynamically.

---

## Understanding HTML Definitions

An HTML definition is a blank window where you can enter HTML. Use HTML definitions for storing long strings of HTML text or HTML text that you use repeatedly in your application. Later, you can access the definition dynamically from an HTML area control by using the GetHTMLText PeopleCode function.

See *PeopleTools 8.52: PeopleCode Language Reference*, "PeopleCode Built-in Functions," GetHTMLText.

You can reference HTML definitions from an HTML area control statically or dynamically. However, the most common usage is to reference HTML definitions dynamically.

---

## Creating HTML Definitions

To create an HTML definition:

1. Select File, New.

The New Definition dialog box opens, enabling you to select the definition you want.

2. Select HTML from the scroll area.
3. Click OK to open a new HTML definition.
4. Type or paste the HTML text directly into the open definition window.
5. Select File, Save to save the HTML definition.

The system saves the HTML definition in the current database.

---

## Opening HTML Definitions

To open an HTML definition:

1. Select File, Open to access the Open Definition dialog box.
2. Select *HTML* from the Definition drop-down list box.
3. Click Open to show all of the HTML definitions that are currently stored in the database.

You can also enter a letter or word in the Name edit box to filter the display. For example, entering the letter *H* reveals all of the HTML definitions that begin with *H* in the database.

Naming a group of HTML definitions with the same introductory phrase can help keep them together for easy selection.

---

## Referencing HTML Definitions Dynamically

To reference an HTML definition dynamically:

1. Add an HTML area control to a page definition.
2. Open the HTML Area Properties dialog box.
3. Select the *Constant* option.
4. On the HTML tab, enter the GetHTMLText PeopleCode function for the HTML definition that you want to reference.
5. Click OK.

### See Also

[Chapter 9, "Using Page Controls," Populating HTML Areas, page 249](#)

[Chapter 11, "Creating Page Definitions," Adding Page Controls, page 329](#)

*PeopleTools 8.52: PeopleCode Developer's Guide*, "PeopleCode and PeopleSoft Pure Internet Architecture," Using HTML Definitions and the GetHTMLText Function



## **Part 3**

# **Maintaining Data**

### **Chapter 18**

#### **Constructing File Layouts and Performing Data Interchanges**

### **Chapter 19**

#### **Using Command Line Parameters**

### **Chapter 20**

#### **Using Change Control**



## Chapter 18

# Constructing File Layouts and Performing Data Interchanges

This chapter provides an overview of file layouts and discusses how to:

- Construct file layouts.
- Specify file layout, record, and field properties.
- Perform data interchanges.
- Produce sample file layouts.

---

## Understanding File Layouts

This section discusses:

- File layouts.
- Field formats in a flat file.
- Fixed-position (FIXED) format files.
- Comma-separated value (CSV) format files.
- XML-tagged (XML) format files.

## File Layouts

A file layout is a definition that represents the data structure of a flat (text) file to be processed. When reading from or writing to a flat file, PeopleCode can use a file layout as a template to identify and correctly process the records and fields of the file. File layouts work with hierarchical and nonhierarchical data, and they can handle files that combine data records with nondata (audit or control) records.

To access data in a file, you do not have to create a file layout. PeopleTools supports reading and writing to plain text files, as well as to files that have a format based on a file layout:

- If the file is a plain text file, you use text strings to read or write data.
- If the file is based on a file layout, you can use text strings, rowsets, or record objects.

Using a file layout greatly simplifies reading, writing, and manipulating hierarchical transaction data with PeopleCode. It can facilitate transactions between a PeopleSoft application and a third-party system when the third-party system does not support PeopleSoft Integration Broker or component interfaces.

---

**Note.** Do not create or save files in Microsoft Excel and then use those files for file processing.

File layouts do not have any related-language processing abilities.

---

### ***Applying File Layouts to Data Interchanges***

In addition to manipulating transaction data, you can employ file layouts to move data between a PeopleSoft database and external systems (data interchange) by using flat files as the transmission medium. File layouts enable you to:

- Export hierarchical PeopleSoft data to several flat file formats.
- Map incoming hierarchical file data directly to PeopleSoft tables.
- Preview and troubleshoot the input data and its formatting before importing it.
- Automatically generate the Application Engine and PeopleCode programs that are needed to perform data import.
- Use batch processes to perform large-volume data imports and exports.

### **See Also**

Chapter 18, "Constructing File Layouts and Performing Data Interchanges," Performing Data Interchanges, page 521

*PeopleTools 8.52: PeopleCode API Reference, "File Class"*

## **Field Formats in a Flat File**

A flat file, in the simplest sense, is a collection of fields in text format. The file must be formatted in a way that enables the PeopleCode to locate each field. PeopleSoft file layouts support three formats:

- **FIXED:** Each field has a starting position and a length that together specify its location in the file.  
This format is the default for new file layouts.
- **CSV:** Fields are located in the file by their sequence, separated by commas.
- **XML:** Fields are located by the named XML tags that surrounding them, not by their position or sequence within a record.

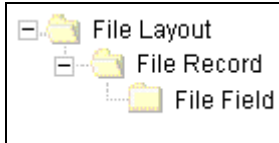
To preserve relationships between fields, you group fields into logical collections. In relational databases, these collections are records. Each line within a file is a collection of fields.

---

**Note.** With some file formats, the logical concept of a record may actually span multiple physical lines, but the concept of field collections remains.

---

The conceptual structure of all three file formats is represented in a file layout as follows:



File layout structure

A file layout is a collection of file records, which in turn is a collection of fields, each of which has a describable location.

---

**Note.** To avoid confusion with the standard terms *record* and *field*, when working with a file layout we refer to a collection of fields as a *file record* and to fields as *file fields*.

---

## FIXED Format Files

FIXED format files are the most common type of flat file that is currently processed by PeopleSoft EDI Manager. Almost all electronic data interchange (EDI) type processing uses this file type where each data element is oriented by a fixed, or column-dependent, position within the file.

The following table describes the properties of FIXED format files:

<i>Property</i>	<i>Description</i>	<i>PeopleSoft EDI Manager Equivalent</i>
File Layout Format	File format (FIXED).	None.
File Record ID	A group of numbers that can be used to identify the file record.	RowID.
ID Start Position	The column starting position of the file record ID.	Treated as a field within each map.
(Record) ID Length	The length of the file record ID.	Treated as a field within each map.
File Record Name	A user-specified name for the file record.	PeopleSoft record name.
File Field	A user-specified name for the file field.	PeopleSoft record field name.
(Field) Start Position	The column starting position of the file field.	Starting position.

<i>Property</i>	<i>Description</i>	<i>PeopleSoft EDI Manager Equivalent</i>
Field Length	The length of the file field.	Length of the field.
Field Format	The formatting options for both inbound and outbound field processing.	Based on the field type.

### **Considerations for FIXED Format Files**

Consider the following points when working with FIXED format files:

- Be careful when you change the length or starting position of any file fields or when you insert a new file field between two existing ones.

You might overlay fields, with unpredictable results.

- When you insert a record into a file layout, fields of the Long type are converted to the Character type with a length of 0.

You must set the field length to a number greater than 0 before you can save the file layout.

## **CSV Format Files**

In this type of file, each data element is surrounded with a separator, a delimiter, or both. You can use file record IDs to determine the source table and the target table; however, in most cases this type of file contains homogenous records.

In the following example, the qualifier is a double quote (") and the delimiter is a comma (,):

```
"NAME " , "ADDRESS " , "PHONE "
```

File layout definitions store the file record ID (when used) and the relative sequence number of the field. In the preceding example, "PHONE" is sequence number 3.

The following table describes the properties of CSV format files:

<i>Property</i>	<i>Description</i>	<i>EDI Manager Equivalent</i>
File Format	File format (CSV).	None.
File Record ID	A group of numbers that can be used to identify the file record.	RowID.
ID Sequence Number	The sequence number of the field that contains the file record ID.	Treated as a field within each map.

<i>Property</i>	<i>Description</i>	<i>EDI Manager Equivalent</i>
Qualifier	The character that surrounds a field to mark its extent. This character can be set at the file layout, file record, or file field level.	Delimiter.
Delimiter	The character used to separate fields from each other. This character can be set at the file layout or file record level.	Separator.
File Record Name	A user-specified name for the file record.	None.
File Record Field	A user-specified name for the file field.	None.
Field Format	The formatting options for both inbound and outbound field processing.	Based on the field type.

### ***Considerations for CSV Format Files***

Consider the following points when working with CSV format files:

- Both the qualifier and the delimiter accept a blank as a valid value.
- If a field is null, you do not have to use qualifiers.

In the following example, Field2 is null:

```
Field1,,Field3,Field4. . .
```

- The system automatically pads the decimal places with zeros when amounts do not have values beyond the decimal point.

For example, 100 is automatically written as 100.00.

### ***Parsing Support for CSV Format Files***

The following list summarizes parsing support in preview and runtime for all CSV file types:

- All blank lines are ignored.  
In the past, blank lines came in as empty records.
- All empty fields are returned as empty strings or zeros.
- Fields, including commas or new lines, are surrounded in quotes.

- When a field is in quotes, any embedded quotes must be escaped by two quotes.
- When non-Microsoft Excel format backslashes are embedded within a string, they must be double backslashes to be interpreted as a backslash; otherwise, they are interpreted as an escape character.

### ***Parsing Support for Microsoft Excel CSV Format***

The following table summarizes the differences between Microsoft Excel CSV parsing and non-Microsoft Excel CSV parsing:

<b><i>Item</i></b>	<b><i>Microsoft Excel CSV Parsing</i></b>	<b><i>Non-Microsoft Excel CSV Parsing</i></b>
White space	Leading and trailing white space is significant for strings.	Leading and trailing whites paces are ignored, including converted tabs.
Backslash	Not interpreted as special characters except for new lines.	Interpreted.
Quotes inside quoted string	Double quoted.	Escaped.
Tab	Converted to a single space.	Interpreted as tabs and ignored.

## **XML Format Files**

This type of file contains data that is represented in a hierarchical or tree-type structure. A tag surrounds each data element. A file record tag might group multiple entries.

File layout definitions tie the identifier, along with parent and child relationships, to the file record and file field.

PeopleSoft EDI Manager has no equivalent format.

The following table describes the properties of XML format files:

<b><i>Property</i></b>	<b><i>Description</i></b>
File Format	The XML file format.
File Record ID	The tag name representing the file record.
Field Identifier	The tag name representing the file field.
Field Format	The formatting options for both inbound and outbound field processing.
File Record Name	A user-specified name for the file record.



<i>Property</i>	<i>Description</i>
Field Name	A user-specified name for the file field.

### ***Considerations for XML Format Files***

Consider the following points when working with XML format files:

- The XML input file must contain at least the fields that are specified in the file layout definition that you use.

If the file is missing any fields, the input rowset does not contain any data.

- If the XML input file contains extra fields that are not specified in the file layout definition, then the ReadRowset method ignores the extra fields.
- When you insert a record into a file layout, fields of the Long type are converted to the Character type, with a length of 0.

You must set the field length to a number greater than 0 before you can save the file layout.

- The system automatically pads the decimal places with zeros when amounts do not have values beyond the decimal point.

For example, 100 is automatically written as 100.00.

- When importing XML files, the longest tag name supported is 30 characters. This restriction is only an issue when importing XML files because PeopleSoft Application Designer automatically enforces the limit .

---

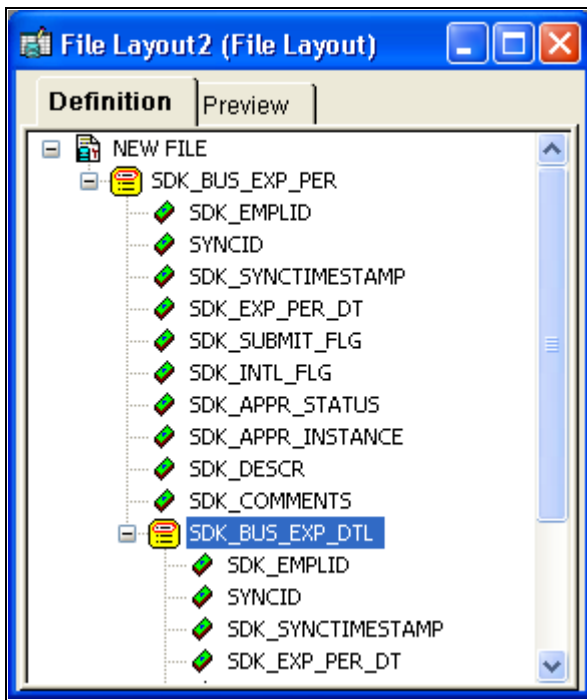
## **Constructing File Layouts**

This section discusses how to:

- Create new file layout definitions.
- Add file records and file fields.
- Reorder file layout nodes.
- Name file layouts, records, and fields.
- Apply field formats.

### **Creating New File Layout Definitions**

In PeopleSoft Application Designer, select File, New, File Layout.



File layout with two records in a hierarchy

A new file layout contains only one element, the root node to which file records are attached. The default root node name is NEW FILE.

Save the new file layout definition. You are prompted to name the file layout, which also becomes the name of the root node.

---

**Note.** The default file layout format is FIXED.

---

### See Also

[Chapter 18, "Constructing File Layouts and Performing Data Interchanges," Specifying File Layout Properties, page 513](#)

## Adding File Records and File Fields

You can add file records and file fields to your file layout in two ways:

- Base them on existing record and field definitions in the database.
- Construct them directly in the file layout without reference to any database records or fields.

A file record that you construct in this way is added as a segment, but the result is an ordinary file record.

You can use both methods in any combination when creating file records and file fields in a file layout definition. Each file record and file field is generically referred to as a node.

---

**Important!** Regardless of the method that you use, file records and file fields exist only as definitions within a file layout and have no connection with any database record or field. Even with file records based on records in the database, a change to the database record definition is *not* reflected in the file layout.

After any update to PeopleTools, you should regenerate all file layout definitions.

---

### ***Using a Segment Instead of a Record***

Suppose that in the file that is provided to you, some of the file records contain new data and must be inserted, while others contain data that updates existing data. You can add a segment with a single field (like `AUDIT_ACTION`) that indicates whether the file record is new or changed. When you process the file, you can use PeopleCode to look at this field and, based on its value, perform the appropriate action.

As another example, suppose that you want to include only two fields from the `PERSONAL_DATA` table in the file. You have two choices: insert the `PERSONAL_DATA` table and manually delete all of the unwanted fields or insert a segment, name it `PERSONAL_DATA`, and then insert the two fields that you want.

### ***Using Segments in Data Interchanges***

If you create a file layout for a data interchange, you can use segments, but each file record must correspond to a record with the same name in the PeopleSoft database, and its file fields must have the same names and data formats as the fields of the database record. The file record can contain a subset of the fields in the corresponding database record. It can also contain a subset or a superset of the fields that are provided in the corresponding file data.

### ***Adding File Records***

To add a file record to the file layout definition, use one of the following methods:

- Drag and drop a database record.

Drag a record definition from the Project window into the Definition window of the file layout, dropping it on the root node or on any existing file record or file field. The new file record appears at the same level as the node that you dropped it on, following all of the other file records at that level. The system inserts all of the constituent fields of the record as well.

- Insert a database record.

With the root node or a file record highlighted, select **Insert, Record**. When you select a record, the new file record appears following the highlighted file record and at the same level. The system automatically inserts all of the constituent fields of the record as well.

- Insert a segment.

With the root node or a file record highlighted, select **Insert, Segment**, and enter a file record name. When you click **OK**, the new file record appears following the highlighted file record and at the same level.

- Insert a child segment.

With a file record highlighted, select **Insert, ChildSegment**, and enter a file record name. When you click **OK**, the new file record appears one level below the highlighted file record but before any others at that level.

---

**Note.** When you add a file record at the root level, it appears immediately below the root node, before all of the other file records.

---

### ***Adding File Fields***

To add a file field to the file layout definition, use one of the following methods:

- Drag and drop a database field.

Drag a field definition from the Project window into the Definition window of the file layout, dropping it on any existing file record or file field. Confirm the field name or enter a different one, and click OK. The new file field appears following the node that you dropped it on.

- Insert a database field.

With a file record or file field highlighted, select Insert, Database Field. Confirm the field name or enter a different one and click OK. The new file field appears following the highlighted node.

- Insert a file field.

With a file record or file field highlighted, select Insert, FileField, and enter a file field name. When you click OK, the new file field appears following the highlighted node.

---

**Note.** Each file field must have a unique name within its parent file record, but file fields in different file records can have the same name.

---

## **Reordering File Layout Nodes**

The file layout definition provides a set of directional arrow buttons in the toolbar, which you can use to reposition any file record within the hierarchy of the file layout or any file field within its parent file record.

The up and down arrows do not change the level of the selected item, just its order among other items at that level. The right and left arrows move the selected item lower and higher in the file layout hierarchy.

---

**Note.** When you reposition a file record in the file layout, its child records are also repositioned, and their child records are repositioned, and so on.

---

## **Naming File Layouts, Records, and Fields**

File layout names can be 30 characters in length, and file record and file field names can be 15 characters in length. All names should follow PeopleSoft naming standards.

Each file record within a file layout must have a unique name, but one file record can have the same name as the file layout. Each file field within a given file record must have a unique name, but file fields in different file records can have the same name.

### ***Using WriteRecord, ReadRowset, and WriteRowset***

If you use the WriteRecord, ReadRowset, or WriteRowset file layout methods for writing to or reading from records, the application record and the file record *must* have the same name, and the application record fields and the file fields *must* have the same names. These methods write only to like-named records and like-named fields within a given record. If you rename a record or a field after you use it to create a file layout definition, you must rename the file record or file field to the exact same name.

In a file layout definition containing more than one record, the system ignores records and fields that are not like-named. Like-named records do not have to contain all of the same fields, and like-named fields do not have to be the same length. Like-named fields should, however, be of the same type.

### **See Also**

[Chapter 6, "Creating Record Definitions," Naming Record Definitions, page 123](#)

## **Applying Field Formats**

The following topics discuss issues you must consider when specifying field formats.

### ***Numbers Format in FIXED Files***

When you write numeric data to a FIXED format flat file, all numbers are right-justified in the file field. Numbers with decimal places specified are written with zeros padding the unused decimal places.

For example, a sequence of records with numbers of varying precision is written this way:

```
001      53.2700BUY
002  2174.0933SELL
003   108.0000SELL
```

### ***Date, Time, and Datetime Field Lengths***

In accordance with the International Organization for Standardization (ISO) 8601 standards, the field lengths for date, time, and datetime fields are fixed in the file layout, regardless of the file format:

- Date fields have a fixed length of 10.
- Time fields have a fixed length of 20.
- Datetime fields have a fixed length of 31.

Datetime fields must use one of these formats.

- CCYY-MM-DD HH:MM:SS
- CCYY-MM-DDTHH:MM:SS (with time separator)
- CCYY-MM-DD HH:MM:SS.ssssss (with fraction of seconds)
- CCYY-MM-DDTHH:MM:SS.ssssss (with time separator and fraction of seconds)

- CCYY-MM-DD HH:MM:SS.ssssss+hhmm or CCYY-MM-DD HH:MM:SS.ssssss-hhmm (with fraction of seconds and universal time zone offset)
- CCYY-MM-DDTHH:MM:SS.ssssss+hhmm or CCYY-MM-DDTHH:MM:SS.ssssss-hhmm (with time separator, fraction of seconds, and universal time zone offset)

---

**Note.** PeopleSoft applications comply with ISO 8601 guidelines for representing date field data. Refer to the ISO 8601 guidelines for more information.

---

See <http://www.w3.org/TR/NOTE-datetime>.

### ***Date Format with the ReadRowset Method***

Single digits in dates in the form MMDDYY or MMDDYYYY must be padded with zeros. That is, if the date in the data is February 3, 2002, then the format must be:

- 02/03/2002
- 02/03/02

The format 2/3/02 is *not* valid.

When using the date format DDMMYY, single digits should also be padded with zeros, and the month value should be all uppercase letters. For example, if the date in the data is February 2, 2003, then the format must be 02-FEB-03. The format 2-Feb-03 is *not* valid.

### **See Also**

*PeopleTools 8.52: PeopleCode API Reference*, "File Class," File Layout Error Processing

---

## **Specifying File Layout, Record, and Field Properties**

This section discusses how to:

- Specify file layout properties.
- Specify file record properties.
- Specify file field properties.

---

**Note.** Some properties are available only for a specific file layout format. For example, a file definition tag is available only for an XML format file. When a property is available only for a particular format, the format is noted in parentheses after the name of the property (such as File Definition Tag [XML]).

---

---

**Note.** Each property that appears in the Preview tab of the File Layout Designer is only another view of the corresponding property of the same name that appears in the Properties dialog box when working with the Definition tab. Changing any property value in one view changes the same property value in another view.

---

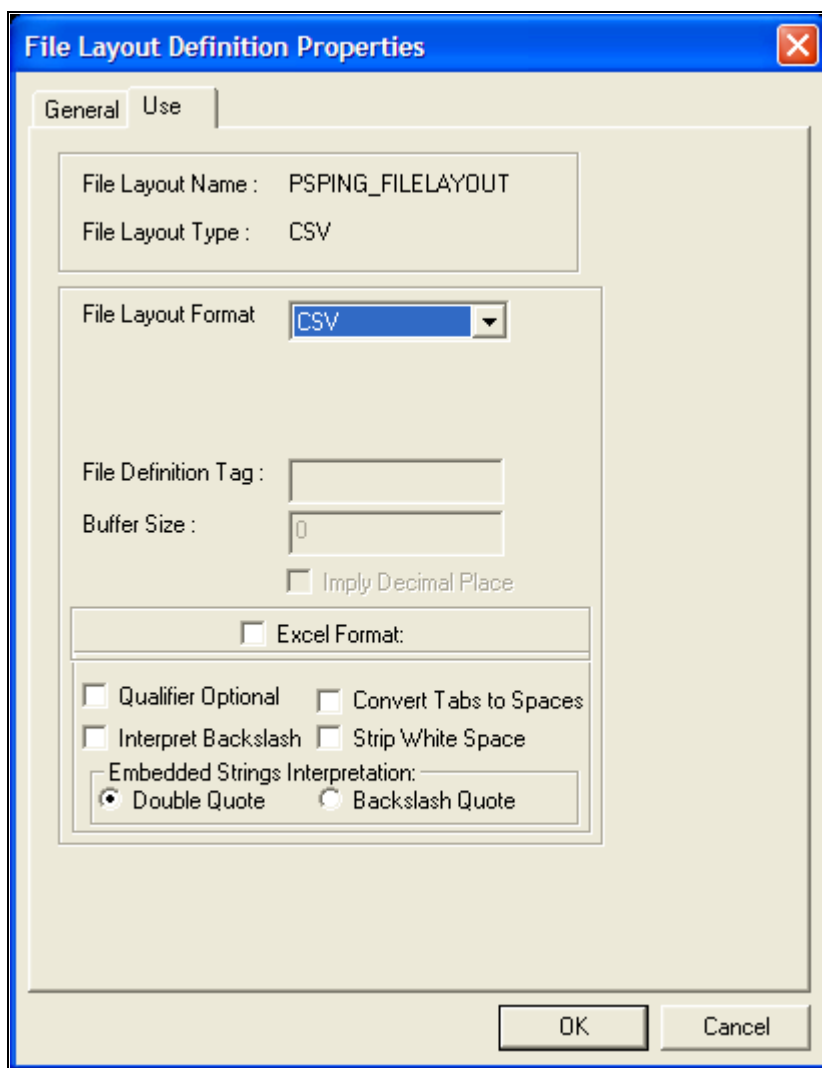
## Specifying File Layout Properties

The File Layout Definition Properties dialog box contains all of the information that is stored at the file layout (root) level.

Access the dialog box by using one of the following methods:

- Select File, Object Properties.
- Press the Alt+Enter keys.
- Double-click the topmost (root) node of a file layout definition.
- Right-click an open file layout and select Data Object Properties.

The General tab of the dialog box contains description information for the file layout. The Use tab contains specific information for the file layout:



File Layout Definition Properties dialog box: Use tab

<b>File Layout Format</b>	Displays the type of file layout. Values are <i>FIXED</i> , <i>CSV</i> , and <i>XML</i> .
<b>File Definition Tag</b>	<p>Enter the XML tag name that is associated with this layout (or transaction). This tag can be 30 characters in length. This tag must be unique in the file layout.</p> <p>This field is active when the file layout format is XML.</p> <hr/> <p><b>Note.</b> If you leave this field blank, the system inserts <code>&lt;start&gt;</code> into the file as the default tag value. Enter any value to prevent the system from inserting the default <code>&lt;start&gt;</code> tag into the file.</p> <hr/>
<b>Buffer Size</b>	<p>Displays the size of the input buffer that is used at runtime.</p> <p>This field is active when the file layout format is XML.</p> <hr/> <p><b>Note.</b> Do not edit this value directly.</p> <hr/>
<b>Imply Decimal Place</b>	<p>Select to automatically add a decimal place based on the decimal value that is specified in the field definition when you import a file into a PeopleSoft database and no decimal place is specified for numeric values in the file.</p> <p>If you select this option, make sure that the length that you specify for the Number field includes a decimal point, even though it is not visible. In this case, the decimal point is counted as a space. For each occurrence of numeric data that is missing a decimal point character in the input file, a space character is also required in the input file; locate the space character either immediately before the numeric characters or immediately after them.</p> <p>This field is active when the file layout format is <i>FIXED</i>.</p>
<b>Excel Format</b>	<p>Select to parse the field by using Microsoft Excel CSV specifications.</p> <p>This field is active when the file layout format is <i>CSV</i>.</p>
<b>Qualifier Optional</b>	<p>Select to parse the field based on delimiters. When a qualifier exists, the field is parsed by using a closing matched qualifier. The system ignores all characters before and after the qualifier.</p> <p>This field is active when the file layout format is <i>CSV</i>.</p> <p>If you do not select this check box, the field <i>must</i> have qualifier pairs. The system ignores all characters before and after the qualifier.</p>
<b>Convert Tabs to Spaces</b>	<p>Select to convert all tabs within the field to spaces. If you do not select this option, the system ignores all tabs.</p> <p>This field is active when the file layout format is <i>CSV</i>.</p> <hr/> <p><b>Note.</b> If the Strip White Space option is also selected, the system removes trailing and leading tabs.</p> <hr/>



**Interpret Backslashes**

Select to interpret backslashes within field data as escape characters.

The following characters have special meanings:

- \t represents a tab.
- \n represents a new line.

If you do not select this check box, the system considers backslashes to be regular text data.

This field is active when the file layout format is CSV.

**Strip White Space**

Select to remove all leading and trailing white space within field data from the output data.

This field is active when the file layout format is CSV.

---

**Note.** If you select this option, the system removes trailing and leading tabs, regardless of whether you selected the Convert Tabs to Spaces option.

---

**Double Quote**

Select to interpret two double quotes (" "), as one double quote (") in output data.

For example, if you enable this option, abc " "def " is interpreted as abc"def .

This field is active when the file layout format is CSV.

**Backslash Quote**

Select to interpret backslash quotes (\") as one double quote (") in output data.

For example, if you enable this option, " abc\"def " is interpreted as abc"def .

This field is active when the file layout format is CSV.

**See Also**

Chapter 18, "Constructing File Layouts and Performing Data Interchanges," Understanding File Layouts, page 501

## Specifying File Record Properties

The File Layout Segment Properties dialog box contains information that is stored at the file record level:

The image shows a dialog box titled "File Layout Segment Properties" with a "Use" tab selected. The dialog contains the following fields and values:

Property	Value
File Layout Name	PSPING_FILELAYOUT
File Layout Type	CSV
File Record Name	TP_TST_CASE_VW
ID Seq No	1
Max Rec Length	106
File Record ID	
ID Start Position	0
ID Length	0
Default Qualifier	
Seg Terminator	
Field Delimiter	Comma
Record Tag	PTP_TST_CASE_V
Record Description	

At the bottom of the dialog are "OK" and "Cancel" buttons.

File Layout Segment Properties dialog box: Use tab

---

**Important!** You must manually synchronize the file layout segment properties with the related properties of any associated record definition. Unexpected results can occur when the properties are not synchronized. These properties are most likely to be unsynchronized immediately after an upgrade.

---

Access the dialog box using one of the following methods:

- Double-click the file record node.
- Right-click the file record node and select Selected Node Properties.

#### File Record Name

Enter a file record name that is associated with this file record. This name is used to access the file record from PeopleCode. Every file record in a file layout must have a unique name.

<b>ID Seq No. (CSV)</b> (identification sequence number)	Enter a sequence number for the field that contains the file record ID.
<b>Max Rec Length</b> (maximum record length)	Displays the default maximum length of the combined field sizes of the record. This value is automatically updated.
<hr/> <b>Warning!</b> Any inbound or outbound data is truncated beyond this value. <hr/>	
<b>File Record ID</b>	Enter a number that uniquely identifies the file record in the file layout. You can use this number for processing the file. This number is automatically written to the file if you use the WriteRecord or WriteRowset methods and if the file type is FIXED or CSV.
<b>ID Start Position (FIXED)</b>	Enter the column or starting position in the file record where the file record ID starts.
<b>ID Length (FIXED, CSV)</b>	Displays the length of the file record ID. This number is automatically generated when you enter the file record ID.
<hr/> <b>Note.</b> Do not edit this value directly. <hr/>	
<b>Default Qualifier (CSV)</b>	Enter a qualifier that is used for the file record ID and as the default for fields when no field qualifier is specified. This value overrides the definition qualifier that is specified in the File Layout Definition Properties dialog box. When you first create a file layout, this property is blank.
<b>Seg Terminator (FIXED, CSV)</b> (segment terminator)	Enter characters to be placed at the end of each segment when the layout is written to a file.
<b>Field Delimiter (CSV)</b>	Enter a delimiter that is used for all fields in the file record. This value overwrites the definition delimiter that is specified on the File Layout Definition Properties dialog box.
<b>Record Tag (XML)</b>	Enter an XML tag name for this file record. The default value is the file record name.
<hr/> <b>Note.</b> Although each record name in a file layout must be unique, record tags do not have to be unique. <hr/>	
<b>Record Description</b>	Enter a description of the record for documentation purposes only.

## Specifying File Field Properties

The File Layout Field Properties dialog box contains information that is stored at the file field level:

**File Layout Field Properties**

Use

Layout Name: PSPING\_FILELAYOUT      Sequence No:     

Layout Type: CSV      1

Field Name: PTP\_TEST\_CASE\_ID      ☐ Trim Spaces      ☐ Pad Field

Start Position: 1      Length: 30      Field Type: Character      ☐ Upper Case

Qualifier:      Decimal: 0

Field Tag: PTP\_TEST\_CASE\_ID      Propagate: <<< 0 >>>

Strip Characters:      Field Description:

Default Value:

Field Inheritance: Record: DO-NOT-INHERIT      Field:

OK      Cancel

File Layout Field Properties dialog box: Use tab

**Important!** You must manually synchronize the file layout field properties with the related properties of any associated field definition. Unexpected results can occur when the properties are not synchronized. These properties are most likely to be unsynchronized immediately after an upgrade.

Access the dialog box using one of the following methods:

- Double-click the file field node.
- Right-click the file field node and select Selected Node Properties.

Most individual properties are usable by all field types. However, some are specific to a particular field type, for example, the Upper Case check box is applicable to character fields only, the Date Separator field is applicable to date fields only, and so on. The preceding dialog box shows the properties for a character type of field. The following descriptions include all possible properties.

**Date Format (Date)**      Select a date format, such as MMDDYY, DDYYMM, and so on.

<b>Date Separator (Date)</b>	Enter a character that is used to separate date values. The default value is /.
<b>Decimal</b>	Enter the number of decimal positions to the right of the decimal point. This property is valid only for fields that are defined as number or signed number. <hr/> <b>Note.</b> Only 31 characters plus a decimal point are allowed. <hr/>
<b>Default Value</b>	Enter the default value to be used by a call to ReadRowset when the corresponding data is imported from a file via a file layout definition. <hr/> <b>Note.</b> You cannot use the system variables related to date and time (for example, %DATE, %TIME, %DATETIME) to specify the value of the default-value property of a file layout field.  The value of the default-value property of a file layout field has no relationship to the value of the default-value property of any corresponding record field (one having the same record name and field name). When data is imported from a file via a file layout definition, the value used for a file layout field that is empty in the input file is determined by the default value of the corresponding file layout field, not the default value of any corresponding record field (even if the value of the default-value property of the file layout field is empty and even if that record field is required). <hr/>
<b>Field Description</b>	Enter a description of the field for documentation purposes only.
<b>Field Inheritance</b>	Optionally select a parent file record and field from which the current field value is to be inherited. If you write to a file, the value is written only in the parent file record, not in the child (inheriting) file record; the value is not written more than once to a file. If no value is present in the parent field, then the system uses the default value that is specified here.  For example, the following file sample shows both the EMPLID (8113) and EFFDT (08/06/1999) written only once to a file, though these fields are repeated in the third file record (with a file record ID of 102):  <pre> 100 8113 Frumman,Wolfgang 101      08/06/1999      000001      219 Going to London office 102                                100 000015 I 08/06/1999 102                                200 000030 I 08/06/1999 102                                300 000009 I 08/06/1999 102                                400 000001 I 08/06/1999 102                                500 000011 I 08/06/1999 </pre>
<b>Field Name</b>	Enter the name that is associated with this file field. Use this name when accessing the file field from PeopleCode. Every field within a file record must have a unique name; however, two different file records can contain the same file field.

<b>Field Tag (XML)</b>	<p>Enter an XML tag name to be used around the field. The default value is the name of the field.</p> <hr/> <p><b>Note.</b> Although each field name in a file record must be unique, each field tag does not have to be unique.</p> <hr/>
<b>Field Type</b>	Select the data type of the file field.
<b>Length</b>	<p>Enter the maximum number of characters in this field.</p> <hr/> <p><b>Note.</b> You are allowed only 32-character precision for number and signed number fields; that is, a total of 32 characters both to the right and left of the decimal. Other fields, such as character fields, can be longer.</p> <p>You cannot set a field length for Date, Time, and Datetime type fields. These field lengths are automatically set to the ISO standards for such fields.</p> <hr/>
<b>Pad Field (CSV)</b>	Select to enable field padding. The field will be padded with space characters up to the value of the length property. Character data will be left justified, numeric data right justified.
<b>Propagate (FIXED)</b>	If a field position or length is changed, enter an amount here to increment (positive number) or decrement (negative number) the current field and all fields before it (<<<) or after it (>>>).
<b>Qualifier (CSV)</b>	Enter a qualifier for the field, that is, the character that surrounds this field, separating it from other fields. Specifying this value overwrites the value that is specified in the file layout properties and file record properties.
<b>Start Position (FIXED)</b>	<p>Enter the starting position (column) of the field within the file record.</p> <hr/> <p><b>Important!</b> If you specify a start position for a field that overwrites a previous field, no data is written to the file. Use Propagate to change the start positions for your file fields.</p> <hr/>
<b>Strip Characters</b>	<p>Specify any characters to be removed from the input buffer. Use this field to preprocess input strings. For example, if a field in the input file contains hyphens but you want to remove the hyphens before processing the field, you can enter a hyphen here and it will be stripped out while being read. You can specify more than one character to be stripped out. Be sure to not separate the strip characters. For example, the following code strips out all semicolons and hyphens:</p> <pre>; -</pre> <p>The following code strips out all semicolons, hyphens, and spaces:</p> <pre>; -</pre>

**Trim Spaces**

Select to remove trailing spaces only from an input string. This option is active only for FIXED and CSV formats.

This option is different from the Strip Characters field, which removes all spaces from the entire input field if you specify a space.

**UpperCase (Char)**

Select to convert lowercase text to uppercase during inbound processing. Use this option primarily when customer data is in lowercase and the PeopleSoft software requires the data to be in uppercase.

---

**Note.** PeopleSoft applications do not accept numbers that contain thousand separators, as in "," or ".". For example, if you exchange data with Microsoft Excel, you need to disable the "Use 1000 Separator (,)" property.

---

**See Also**

Chapter 4, "Creating Field Definitions," page 63

---

## Performing Data Interchanges

This section provides an overview of the import process and discusses how to:

- Preview input data.
- Generate and run the import program.
- Export data.

## Understanding the Import Process

To help you troubleshoot and import flat file data, the file layout definition provides a data preview page. It can also generate an Application Engine program with the associated PeopleCode that is necessary to import the data.

---

**Note.** The Application Engine program that the system automatically generates should be viewed as containing sample PeopleCode that you need to review and modify, especially when a single file layout involves multiple records.

---

***Data Import Activities***

Importing data by using a file layout requires the following sequence of activities:

1. Provide the import data in a properly formatted flat file.

Each record in the file must correspond to a record with the same name in the PeopleSoft database, and its fields must have the same names and data formats as the fields in the database record. Each record in the file must end with a new line character.

2. Create a file layout definition to match the record and field structure of the data.

Insert the appropriate record definitions into the file layout, and then reposition the file records and file fields to match the record and field positions in the file.

3. Preview and troubleshoot the input data format and content.
4. Generate the data import Application Engine program and PeopleCode.
5. Run the Application Engine program to import the data.

### ***Field Inclusion and Exclusion***

The fields in the records of the data file can be a subset of the fields in the database record. Define your file layout with only the file fields that you expect to receive.

The fields in the records of the data file can be a superset of the fields in the database record; you must define your file layout to suppress or ignore the extra fields. For FIXED files, do not define a field at the corresponding position in the file layout definition. For CSV files, the file layout must have the same number of fields in each record as there are in the corresponding file record; for each field that you do not want to import, define a field in the file layout at that position that does not correspond to any field in that database record. For XML files, the system automatically ignores any extra fields.

Your data file can contain a subset or a superset of the records that are defined in the file layout. The system imports only a file record with a matching file record ID in the file layout.

### ***Record Hierarchy***

In theory, you can ignore rowset hierarchy when importing file data, because the PeopleSoft database stores each record independently of the others, and the import process does not use rowsets. However, many records are designed with hierarchical dependencies in mind. The input file might omit inherited field values or order the data records in a way that reflects such dependencies.

If the input file omits inherited field values, make sure that the records in the file layout of the inheriting fields are children of the ones from which they inherit their values and set up Field Inheritance appropriately.

If the records to be imported contain key fields that reflect a rowset hierarchy, they might be in an order in the file that also reflects the hierarchy. Make sure your file layout reflects that hierarchy as well.

---

**Important!** The completed file layout must have exactly one file record at the root level; all other file records must be below that level.

---

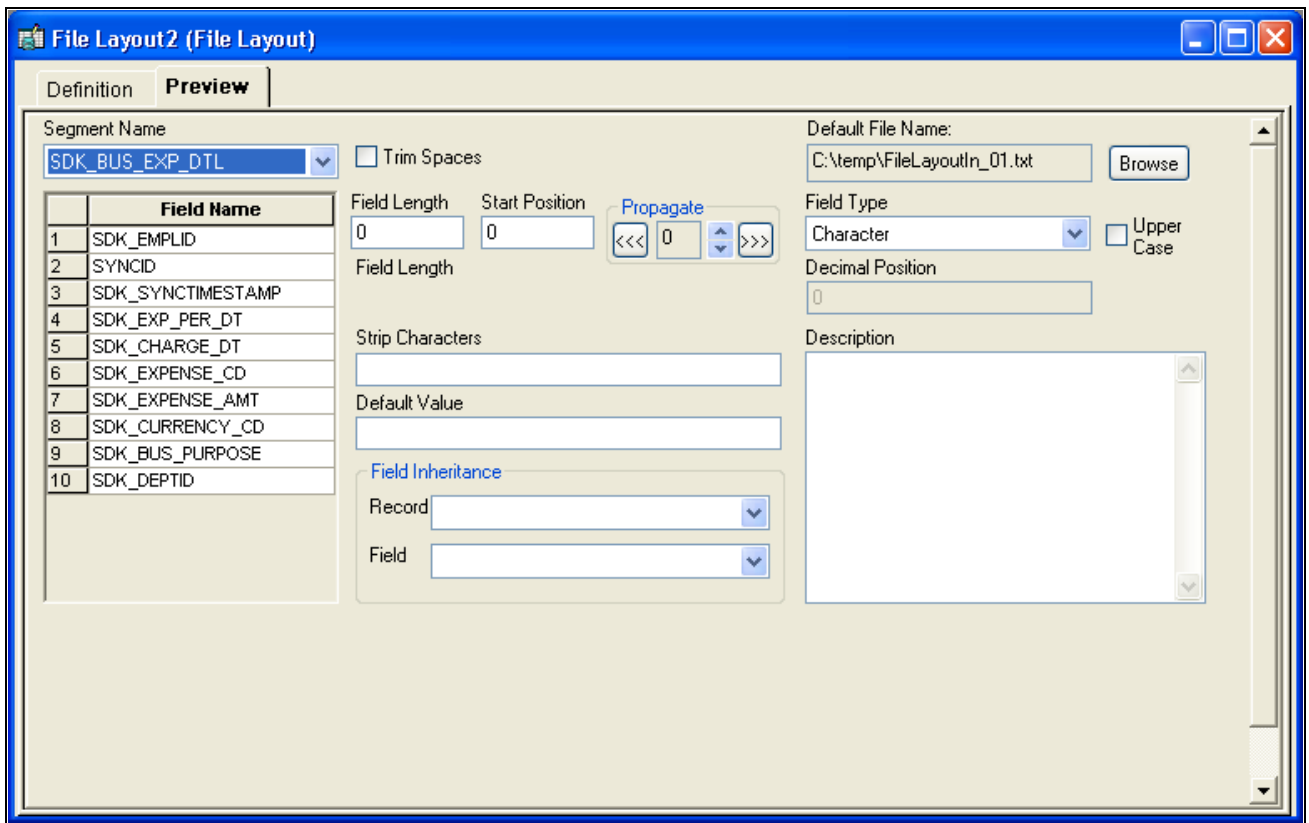
### ***See Also***

Chapter 18, "Constructing File Layouts and Performing Data Interchanges," Reordering File Layout Nodes, page 510

## **Previewing Input Data**

In PeopleSoft Application Designer, select File, New, File Layout, Preview.





File layout preview

The Preview tab displays no information initially.

---

**Note.** With an XML file, you cannot preview the data, although you can generate and run an import program.

---

To preview input data:

1. Click Browse and select the import file.

The name of the import file appears in the Default File Name text box.

2. Select a file layout segment from the Segment Name drop-down list box.

The file layout fields corresponding to the selected segment appear in the list below the segment name. This list comes from the file layout definition, and it does not depend on your selecting a valid import file.

3. Click the Refresh button on the toolbar to refresh the preview data.

At the bottom of the window, the first five rows of the selected segment in the selected import file are available for preview in the grid.

You can also select View, Automatically Read Preview Data. While this menu item is selected, the file layout refreshes the preview automatically; when it is cleared, you need to click the Refresh Preview Data button to see the effects of changes you make to the property settings or data.

---

**Note.** Preview file data is available only for FIXED and CSV file layouts. The preview data appears only if you selected an import file, if the file format matches the format specified for the file layout definition, and if it contains a file record ID that matches the file record ID of the selected segment.

---

#### 4. Select one of the fields in the Field Name list box.

The properties of the field that you select appear in the appropriate display fields. These properties are actually the file layout field properties of the field; you can change all field property settings from this view.

---

**Note.** Only the field properties that are appropriate to the file layout format that you specified in the File Layout Definition Properties dialog box are visible in this view. For example, the Start Position and Field Length fields are available only for a FIXED format file layout.

---

### **Examining the Input Data for Errors**

The following table shows examples of input data errors:

<b>Symptom</b>	<b>Possible Reason</b>	<b>Solution</b>
The preview grid does not appear.	The input file format does not match the file layout format that you specified in the File Layout Definition Properties dialog box.  A CSV file does not use the definition qualifier you specified in the File Layout Definition Properties dialog box.	Change the file layout format to match the input file format.  Change the definition qualifier to match the one used in the input file.
Only the first column of the preview grid is populated.	A CSV file does not use the definition delimiter you specified in the File Layout Definition Properties dialog box.	Change the definition delimiter to match the one used in the input file.
The preview grid appears for some records but not for others.	The record IDs of the file for the missing records do not match the file record IDs specified in the File Layout Segment Properties dialog box.	Specify file record IDs that match the input file records.
Data for a field appears truncated in the preview grid.	With a FIXED file, the field length that you specified in the File Layout Field Properties dialog box is too short to accommodate the field data.	Increase the field length to accommodate the input data.
A field appears to start in the middle of the data.	With a FIXED file, the start position that you specified in the File Layout Field Properties dialog box is too great to include the start of the field data.	Decrease the start position and adjust the field length to match the input data start position and length.

## Generating and Running the Import Program

This part of the process imports the file data.

To generate and run the import program:

1. After you preview the file layout and review the input data, click the AE button on the toolbar.

The system generates the Application Engine import program with its associated import PeopleCode.

2. Enter a name for the Application Engine program and click OK.

The system automatically saves the program, which is ready to run as soon as its definition appears in PeopleSoft Application Designer.

3. Click the Run Program button on the toolbar.

The database imports the file data.

## Exporting Data

The method that you use to export data from the PeopleSoft system depends on the requirements of the target application. To export data to a flat file, you create a file layout definition and then write PeopleCode to transfer the data to a file. You can initiate the PeopleCode from the Application Engine or from any event. It should populate text strings, rowset objects, or record objects, and it should apply the WriteRecord or WriteRowset method from the File class to transfer the data to the file by using the file layout definition to position the records and fields as required by the target application.

---

**Note.** To generate valid XML files, be sure to use the Close method in the File class when you finish writing to the file.

---

---

**Important!** As is the case with imports, the completed file layout that you use to export the data must have exactly one file record at the root level; all other file records must be below that level.

---

### See Also

*PeopleTools 8.52: PeopleCode API Reference, "File Class"*

---

## Producing Sample File Layouts

This section discusses how to:

- Create file layout definitions.
- Adjust file layout properties.
- Insert segments and fields.

## Creating File Layout Definitions

The following example shows how to create a file layout to be used with the QE\_ABSENCE\_HIST record:

QE_ABSENCE_HIST (Record)							
Record Fields				Record Type			
	Num	Field Name	Type	Len	Format	Short Name	Long Name
	1	QE_EMPLID	Char	11	Upper	ID	EmplID
	2	QE_ABSENCE_TYPE	Char	3	Upper	Type	Absence Type
	3	QE_BEGIN_DT	Date	10		Begin Date	Begin Date
	4	QE_RETURN_DT	Date	10		Return Dt	Return Date
	5	QE_DURATION_DAYS	Nbr	3		Days	Duration (Days)
	6	QE_DURATION_HOURS	Nbr	1.1		Hours	Duration (Hours)
	7	QE_REASON	Char	30	Mixed	Reason	Reason
	8	QE_PAID_UNPAID	Char	1	Upper	Paid/Unpd	Paid/Unpaid
	9	QE_EMPLOYER_APPROV	Char	1	Upper	Approved	Employer-Approved
	10	QE_COMMENTS	Long	0		QE_COMMENTS	QE_COMMENTS

QE\_ABSENCE\_HIST record definition

For simplicity, suppose that each row in the FIXED format file has the following structure:

```

888 A
000 8001 VAC 1981-09-12 1981-09-26 14 .0 P Y
888 A
000 8001 VAC 1983-03-02 1983-03-07 5 .0 P Y
888 A
000 8001 VAC 1983-08-26 1983-09-10 13 .0 P Y
888 A
000 8516 MAT 1986-06-06 1986-08-01 56 .0 P Y
888 C
000 8516 SCK 1988-08-06 1988-08-07 1 .0 P Y
888 A
000 8516 VAC 1987-07-14 1987-07-28 14 .0 P Y
888 A
000 8553 JUR 1990-12-12 1990-12-17 5 .0 Local Jury Duty P N
888 A
000 8553 MAT 1992-02-20 1992-10-01 224 .0 Maternity Leave U N
888 A
000 8553 MAT 1994-08-19 1995-03-01 194 .0 Maternity U Y
888 A
000 8553 PER 1993-04-15 1993-04-19 4 .0 U N Personal
Day required
888 C
000 8553 SCK 1987-01-28 1987-01-30 2 .0 Hong Kong Flu P N
888 A
000 8553 SCK 1988-08-02 1988-08-03 1 .0 Sick P N
888 A
000 8553 SCK 1995-09-12 1995-09-13 1 .0 P N
888 C
000 G001 MAT 1991-07-02 1991-09-28 88 .0 3-month Maternity P Y Maternity will
be paid as 80% of Claudia's current salary.

```

000 is the file record ID for ABSENCE\_HIST, and each field appears in the same order as in the ABSENCE\_HIST database record. 888 is the file record ID for an extra segment called CHANGE\_ACTION, which contains an AUDIT\_ACTION field with the following meanings:

- A: Row inserted.
- C: Row updated, but no key fields changed.

---

**Note.** The end of file (EOF) character must be on a separate line and not on a line containing data for any incoming file, regardless of the file type. Each data line needs to be terminated with an end of line (EOL) character, which is different than an EOF character.

---

To create a file layout definition:

1. Use the QE\_ABSENCE\_HIST record definition as the template for the file layout.

Create a new file layout and then drag the QE\_ABSENCE\_HIST record into the open file layout.

2. Save the file layout with the name ABS\_HIST.

The name of the first node changes from NEW FILE to ABS\_HIST.

## Adjusting File Layout Properties

In the file layout definition, open the file layout, record, or field properties as appropriate for each step in the following procedure.

To adjust file layout, record, and field properties:

1. Change the file layout properties.
  - a. Double-click the topmost node in the file layout, ABS\_HIST, to display the file layout properties.
  - b. Enter a short and long description of the file layout you are creating. For this example, you are creating a FIXED file layout, so you do not need to make any changes on the Use tab.
2. Change the file record properties.
  - a. Double-click the QE\_ABSENCE\_HIST file record to display its properties. Enter a record ID of 000, and a starting position of 1. The ID length is automatically set.

When you click OK, you see a message asking if you want to increment the start positions for all fields.

- b. Click Yes to automatically increment the start position numbers for every field to take the length of the file record ID that you just added into account. If you do not click Yes, you must manually increment the start position for all fields.

You just created the file record ID for the QE\_ABSENCE\_HIST record:

000	8001	VAC	1981-09-12	1981-09-26	14	.0	P	Y
000	8001	VAC	1983-03-02	1983-03-07	5	.0	P	Y

### 3. Change the file field properties.

When a record definition is used as a template for a file layout, the default starting position for each field is based on the order in which it appears in the record as well as its length.

- a. Double-click the QE\_EMPLID file field to display its properties.

The start position is automatically incremented to 4 because the file record ID is 3 characters long. However, in the example, an extra space is between the end of the file record ID and the first field. Therefore, you need to change the start position of this field and all of the fields that follow this field to increment the starting position of this field and all fields following this field by 1. To do this:

- b. Click the up arrow under Propagate to change that number from 0 to 1.
- c. Click the button with the arrows pointing right (>>>).

### 4. (Optional) Adjust other fields.

The last field (QE\_COMMENTS) has a length of 0 because it is based on a Long field type.

---

**Note.** When a Long field type is inserted into a file layout, the system converts it to a Character field type with a length of 0.

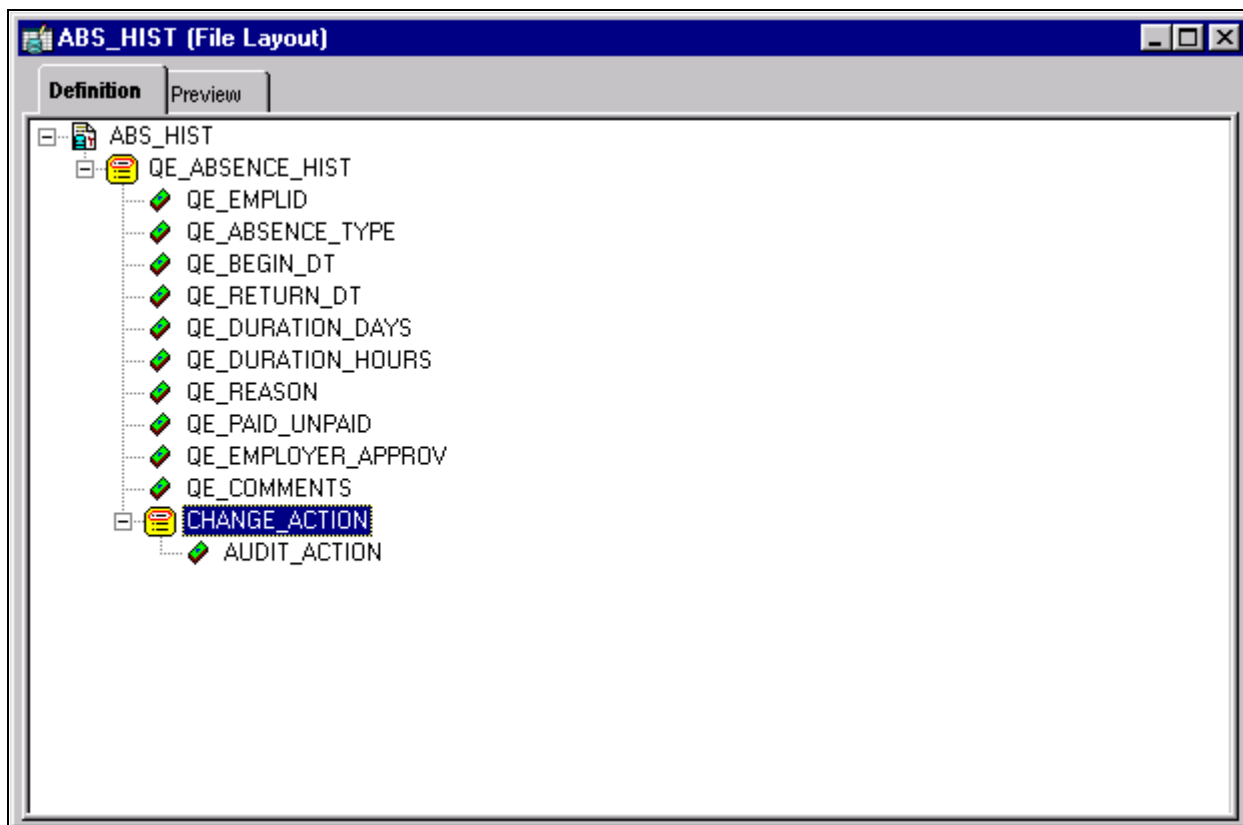
---

Because the format for this file layout is FIXED, you have to change the field length of that file field so that it is long enough to accommodate the data that you expect from the file.

You do not have to propagate this change because this is the last field in the record.

## Inserting Segments and Fields

Open the file layout definition based on the ABS\_HIST record:



File layout with new segment and field added

---

**Note.** You will insert a segment that is a sibling, that is, it has the same level as the QE\_ABSENCE\_HIST record.

---

To insert a segment and then a field into that segment:

1. Select Insert Segment from the pop-up menu or select Insert, Segment to insert a segment.

The Insert New Segment dialog box appears. This dialog box is identical to the File Layout Segment Properties dialog box.

2. On the Insert New Segment dialog box:
  - a. Enter the file record name, file record ID, and ID start position.
  - b. Click the OK button to insert the segment.

---

**Note.** If you provide a Record File ID when adding a segment to a file layout that is either CSV or FIXED, the system prepends this Record File Id to every row of data that is written.

---

3. Insert a file field.

- a. Select Insert FileField from the pop-up menu.

The Insert New Field dialog box appears. It is identical to the File Layout Field Properties dialog box.

- b. Enter a field name, start position, and field length, and then click the OK button.

---

**Note.** The start position is not automatically set when you add a file field to the file record. For this example, the AUDIT\_ACTION field requires a start position of 5.

---

4. Select File, Save, or click the save icon in the toolbar to save your changes to the file layout.

Now that you have created and saved a file layout, you must use PeopleCode to access the data. File layouts rely solely on PeopleCode as the engine behind the actual data access and movement.

**See Also**

*PeopleTools 8.52: PeopleCode API Reference, "File Class"*



## Chapter 19

# Using Command Line Parameters

This chapter provides an overview of PeopleSoft Application Designer command line parameters and discusses how to:

- Start the PeopleSoft Application Designer Command Line Interface.
- Build projects.
- Copy definitions.
- Compare definitions for upgrade using the command line.

---

## Understanding PeopleSoft Application Designer Command Line Parameters

PeopleSoft Application Designer offers a variety of command line parameters that you can use to control the database to which it connects. By using these parameters, you can automatically navigate to the part of the system that you need.

Use the values from the following tables to specify certain parameters.

### ***Syntax***

The following command line syntax applies to all actions identified in this chapter:

PSIDE [-parameter value [-parameter value . . .]]

You can include as many or as few parameters as you need.

Each parameter starts with a hyphen (-) or a forward slash (/). The value for each parameter follows the hyphen or slash, separated by zero or more spaces. In general, the value does not need to have quotation marks around it, even if it has internal spaces: the system treats all text following the parameter as part of the value, up to the next parameter or the end of the command line.

---

**Note.** You must enclose a value in quotation marks only when it includes a hyphen or forward slash, or to include leading or trailing spaces. If the value itself includes a quotation mark character, precede the double quote with a backslash (\).

---

**See Also**

*PeopleTools 8.52: Data Management*, "Using PeopleSoft Data Mover," Using the PeopleSoft Data Mover Command-Line Interface

---

## Starting the PeopleSoft Application Designer Command Line Interface

Use the command line syntax and available parameters to start PeopleSoft Application Designer.

**Command Line Syntax**

Use the command line syntax described previously to start PeopleSoft Application Designer.

See [Chapter 19, "Using Command Line Parameters," Understanding PeopleSoft Application Designer Command Line Parameters, page 531.](#)

**Available Parameters**

The available parameters for PSIDE.EXE fall into two general categories:

- Parameters providing login information.
- Parameters setting general options.

This table lists all of the available parameters:

<b>Parameter</b>	<b>Value</b>	<b>Description</b>
-CT	Database type	Enter the type of database to which you are connecting. Valid values are ORACLE, INFORMIX, SYBASE, MICROSOFT, DB2ODBC, and DB2 UDB for Linux, UNIX, and Microsoft (note the spelling of MICROSOFT).
-CS	Server name	Enter the name of the database server for the database to which you are connecting. This setting is required for some database types.
-CD	Database name	Enter the name of the database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.

<b><i>Parameter</i></b>	<b><i>Value</i></b>	<b><i>Description</i></b>
-CO	User ID	Enter the PeopleSoft user ID you log in with.
-CP	Password	Enter the password for the specified user ID. If you omit this parameter the PeopleSoft Signon dialog box appears and prompts the user for the password.
-CI	Connect ID	Enter the ID that is used to connect to the database server.
-CW	Password	Enter the password for the specified connect ID.
-SS	NO	Use to suppress the display of the PeopleSoft splash screen. Any other value causes the splash screen to appear.
-SN	NO	Use to suppress the sound that plays when you sign in to the PeopleSoft system. Any other value causes the sound to play.
-QUIET	(none)	Use to run in quiet mode so that no message boxes appear. The system writes messages to the log file or trace file, depending on whether you also use the -LF parameter. Quiet mode is recommended for use with processes that require no human interaction.
-HIDE	(none)	Use to hide the PeopleSoft Application Designer interface.
-LF	Log file path name	Path to the log file when used in conjunction with the -QUIET parameter. If used, the system writes any messages generated during processing to the log file.
-FP	Filename	(PeopleSoft Data Mover only) Enter the name of the Data Mover script to run.

**Example**

This command line statement starts PeopleSoft Application Designer and signs in to the DEP7TST database:

```
PSIDE -CT MICROSOFT -CS SEPNTDB05 -CD EP7TST -CO VP1 -CP VP1
```

---

## Building Projects

Use the same command line syntax as identified previously to build a project.

See [Chapter 19, "Using Command Line Parameters," Understanding PeopleSoft Application Designer Command Line Parameters, page 531.](#)

Use the command line statement for the project build to:

- Connect the project to the source database.
- Use the build settings from the Windows registry to generate the SQL script for the create or alter processes.

Before running the build command line, verify that:

- The project's system components are created and loaded.
- The build options are set in the Windows registry.

**Available Parameters**

This table lists all of the available parameters for the project build statement:

<i>Parameter</i>	<i>Description</i>	<i>Error Handling</i>
-HIDE	Use to hide the PeopleSoft Application Designer interface. Always use this parameter when performing a command line Copy process.	Required for all project command line processes.
-CT	Database type. Enter the type of the database to which you are connecting (MICROSOFT, ORACLE, SYBASE, and so on).	Required. If you omit this parameter, the last database type is taken from the registry. If it fails, the process stops and error messages are written to the log file.
-CS	Server name. Enter the name of the source database server for the database to which you are connecting.	Required for some database types. If you omit this parameter, the process stops and error messages are written to the log file.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-CD	Database name. Enter the name of the source database to which you are connecting, as you would enter it into the PeopleSoft Signon dialog box.	Required. If you omit this parameter, the process stops and error messages are written to the log file.
-CO	User ID. Enter the PeopleSoft user ID that is needed to sign in to the source database.	Required. If you omit this parameter, the last database type is taken from the registry. If it fails, the process stops and error messages are written to the log file.
-CP	User password. Enter the password for the specified user ID for the source database.	Required. If you omit this parameter, the PeopleSoft Signon dialog box appears and prompts the user for the password. If the password fails, the process stops and error messages are written to the log file if the log file name parameter is specified.
-PJB	Project name. Enter the name of the project to be built. This project should be available in the database before starting the command line project build.	Required. This is the main parameter and is used internally to decide whether the user can build a project. If you omit this parameter and if you supply all of the source database login parameters, the program only starts the application.

### ***Example***

For this example, assume that the:

- Project name is CJR1.
- Database type is Microsoft.
- Database name is CJR810G.
- User ID is PTDMO.
- Password is PTDMO.
- Path name of the SQL script file is c:\temp\psbuild.sql.
- Path name of the log file is c:\temp\psbuild.log in the Windows registry.

For this example, you would enter the following at the command line:

PSIDE.EXE -CT MICROSOFT -CD CJR810G -CO PTDMO -CP PTDMO -PJB CJR\_PRJ -HIDE -QUIET

## Copying Definitions

Use the same command line syntax as identified previously to copy definitions.

See [Chapter 19, "Using Command Line Parameters," Understanding PeopleSoft Application Designer Command Line Parameters, page 531.](#)

Use the command line statement for Upgrade Copy processes to:

- Connect to the source database.
- Connect to the target database.
- Copy the project and its objects from the source database to the target database.

If the same project already exists in the target database, you can set the option to overwrite the older project with the new project.

### Available Parameters

The following table lists all available parameters for the upgrade copy statement:

<i>Parameter</i>	<i>Description</i>	<i>Error Handling</i>
-HIDE	Use to hide the PeopleSoft Application Designer interface. Always use this parameter when performing a command line Copy process.	Required for all project command line processes.
-CT	Source database type. Enter the type of source database to which you are connecting (for example, ORACLE, SYBASE, and so on).	Required. If the process stops, error messages are written to the log file if the log file name parameter is specified.
-CS	Source server name. Enter the name of the source database server for the database to which you are connecting.	Required for some database types. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-CD	Source database name.  Enter the name of the source database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.	Required. If you omit this parameter, the process stop and error messages are written to the log file if the log file name parameter is specified.
-CO	Source user ID.  Enter the PeopleSoft user ID to use to sign in to the source database.	Required. If you omit this parameter, the user ID is taken from the registry. If it fails, the process stops and error messages are written to the log file (if the log file name parameter is specified).
-CP	Source user password.  Enter the password for the specified user ID for source database.	Required. If you omit this parameter, the PeopleSoft Signon dialog box appears and prompts the user for the password. If the password fails, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TS	Target server name.  Enter the name of the target database server for the database to which you are connecting.	Required for some database types and for a Database Copy process (but not used for a Copy To File or Copy From File process). If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TD	Target database name.  Enter the name of the target database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.	Required for a database Copy process (but not used for a Copy To File or Copy From File process). If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TO	Target user ID.  Enter the PeopleSoft user ID to use to sign in to the target database.	Required for a Database Copy process (but not used for a Copy To File or Copy From File process). If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-TP	Target user password.  Enter the password for the specified user ID for the target database.	Required for a Database Copy process (but not used for a Copy To File or Copy From File process). If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-LF	Log file name.  Enter the name of the file in which error messages are logged during the command line Upgrade Copy process.	Not required. If you omit this parameter, a file is created with the specified path and name, and all of the processing and error messages are written to that file.
-PJC	Source project name.  Enter the name of the project that is to be copied from the source database to the target database. This project should be available in the source database before starting the command line Upgrade Copy process to target database.	Required for a Database Copy process. This is the main parameter that is used internally by the executable file to identify an Upgrade Copy process. If you omit this parameter and if you supply all of the source database login parameters, the executable file starts the application.
-PJTF	Copy Project To File project name.  Enter the name of the project to be copied from the source database to the file.	Required for a Copy To File process. Do not include the file path to the project with this parameter. Use -FP to specify the file path.
-PJFF	Copy Project From File project name.  Enter the name of the project to be copied from the file into the source database.	Required for a Copy From File process. Do not include the file path to the project with this parameter. Use -FP to specify the file path.
-FP	Project file path.  Enter the path to the project for a Copy to File or Copy From File process.	Required for a Copy To File or Copy From File process (but not for a Database Copy process). For the Copy From File process, the directory must exist. For the Copy To File process, any project with the same name is overwritten unless you use the -OVW parameter. A directory specification is created if it does not exist. Do not include the name of the project in the file path. Use -PJTF or -PJFF to specify the project name.



<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-OVD	<p>Override dependencies (0 or 1).</p> <p>Enter 1 to override the checks for unapplied dependency incidents during the Copy From File process and to perform the copy even if dependency incidents for the project have not been applied to the source database.</p>	<p>Not required, and not applicable for processes other than the Copy From File process. The default is 0. When you enter the default and if the project's dependency incidents have not been applied to the source database, an error message is written to the log file listing which incidents need to be applied. The Copy From File process also stops. When you enter 1, a warning message is written to the log for unapplied incidents and processing continues.</p>
-CL	<p>Commit limit.</p> <p>Enter the Commit limit (a number greater than 0) for the number of objects to be copied or compared before a commit is issued. Example: -CL 150</p>	<p>Not required. The default is 50 if the user does not set this parameter.</p>
-AF	<p>Audit flags on records (0 or 1).</p> <p>Enter 1 to keep the target audit flags as is. Enter 0 to take them from the source database. Example: -AF 0</p>	<p>Not required. The default is 1.</p>
-DDL	<p>Data definition language (DDL) on records and indexes (0 or 1).</p> <p>Enter 1 to keep the target database DDL flags as is. Enter 0 to take them from the source database. Example: -DDL 0</p>	<p>Not required. The default is 1.</p>
-PPL	<p>Portal registry structures permission list references.</p> <p>Enter 1 to keep the target portal registry permission list references as is. Enter 0 to take them from the source database.</p>	<p>Not required. The default is 1.</p>
-EXP	<p>Export project definition (0 or 1).</p> <p>Enter 1 to export the project definition to the target database when using the Database Copy process. Not applicable for Copy Project to File process.</p>	<p>Not required. The default is 1. Any project with the same name is overwritten with the new project definition.</p>

<b>Parameter</b>	<b>Description</b>	<b>Error Handling</b>
-OBJ	Object types to copy  List the object types as numbers with commas as delimiters. For example, if you choose records and indexes alone for copying, use the appropriate numbers for records and indexes from the Definition Type Selection table. For records, the number is 0, and for the indexes, the number is 1. Example: -OBJ 0,1	Not required. If you omit this parameter, then all of the objects are copied by default.
-RST	Reset done flags (0 or 1).  Enter 1 to reset the done flags before initiating the Copy process. Enter 0 to not reset them. Example: -RST 0	Not required. The default is 1.
-OVW	Overwrite existing project (0 or 1). Enter 1 to overwrite any existing projects with the same name in the specified path during the Copy Project To File process, or in the database in the Copy Project From File process. Enter 0 to not overwrite existing projects. This parameter applies to the Copy Project To and From File processes only.	Not required. The default is 1.
-LNG	Copy languages.  List the language codes with commas as delimiters or enter ALL to select all available languages. For new languages that are not listed in the Language Selection table, use the xlat short name from the LANGUAGE_CD field in the translate table. For example, if you choose English and Spanish for copying, enter the appropriate codes for English and Spanish from the Language Selection table. For English, the code is ENG, and for Spanish, the code is ESP. Example: -LNG ENG,ESP  See <a href="#">Appendix B, "Referencing Selections for Command Line Parameters," Language Selections, page 586.</a>	Not required. If you omit this parameter, then the languages that are already set in the project are used as the default. The COMMON language code refers to nonlanguage attributes that are not translated. Translated attributes can be copied separately from nontranslated attributes. If you do not specify the COMMON code when using -LNG, you copy only the translated language attributes.  Changing languages on the command line changes the values that are stored with the project definition. Subsequent Copy and Compare processes on the modified project use the new values unless they are explicitly changed again in the upgrade options.

<b>Parameter</b>	<b>Description</b>	<b>Error Handling</b>
-CFD	<p>Keep Target ChartField PageField Display Size.</p> <p>(Number 1 or 0). This parameter indicates whether or not the target chartfield page field display size attributes are to be kept, or whether they should be set from source during copy. See Upgrade Options for more information.</p> <p>Enter 1 to keep the target chartfield page field display size attributes.</p> <p>Enter 0 to set these attributes from the source values.</p>	Not required. Default is 1
-CFF	<p>Keep Target ChartField Field Format.</p> <p>(Number 1 or 0) This parameter indicates whether or not the target chartfield field format attributes are to be kept, or whether they should be set from the source during Copy. See Upgrade Options for more information.</p> <p>Enter 1 to keep the target chartfield field format attributes.</p> <p>Enter 0 to set these attributes from the source values.</p>	Not required. Default is 1

### **Example: Database Copy Process**

These are some command line statements for the Database Copy process:

- Copy the project PPLTOOLS from the source Oracle database, PTDMO, to the target database, PTTST.
  - Log process and error messages to c:\temp\copy.log.
  - Set the commit limit to 150.
- Copy the audit flags and record and index DDLs from the source database.
- Copy only records, indexes, pages, queries, and process definitions from the project.
  - Do not reset the done flags before initiating the Copy process.
  - Copy only English and Spanish translations.

This is a command line statement that performs the preceding list of tasks.

```
PSIDE.EXE -HIDE -PJC PPLTOOLS -CT ORACLE -CD PTDMO -CO PTDMO -CP PTDMO -TD PTTST
-TO PTDMO -TP PTDMO -QUIET -LF C:\TEMP\COPY.LOG -CL 150 -AF 0 -DDL 0 -OBJ 0,1,5,10,20 -
RST 0 -LNG ENG, ESP
```

### **Example: Copy To File Process**

These are some command line statements for the Copy to File process:

- Copy the PPLTOOLS project from the Oracle database, PTDMO, by using the PTDMO user ID, to the local directory c:\temp\export.

A directory named PPLTOOLS is created in c:\temp\export.

- Log progress and error messages to c:\temp\copy.log.

This is a command line statement that performs the preceding list of tasks.

```
PSIDE.EXE -HIDE -PJTF PPLTOOLS -FP c:\temp\export -CT ORACLE -CD PTDMO -CO PTDMO -CP
PTDMO -QUIET -LF c:\temp\copy.log
```

### **Example: Copy From File Process**

These are some command line statements for the Copy From File process.

- Copy the PPLTOOLS project to the DB2 UDB for Linux, Unix, and Windows database, PTDMO, by using the user ID PTDMO, from the local directory c:\temp\export.

A directory named PPLTOOLS must exist under c:\temp\export.

- Log progress and error messages to c:\temp\copy.log.

This is a command line statement that performs the preceding list of tasks.

```
PSIDE.EXE -HIDE -PJFF PPLTOOLS -FP c:\temp\export -CT DB2UNIX -CD PTDMO -CO PTDMO -CP
PTDMO -QUIET LF c:\temp\copy.log
```

---

## **Comparing Definitions for Upgrade Using the Command Line**

Use the same command line syntax as identified previously to compare definitions.

See [Chapter 19, "Using Command Line Parameters," Understanding PeopleSoft Application Designer Command Line Parameters, page 531.](#)

Use the command line statement for an Upgrade Compare process to:

- Connect to the source database.
- Connect to the target database.
- Perform a project or database compare (based on parameters).
- Create compare reports in the specified output directory.

## Available Parameters

This table lists all of the available parameters for the upgrade compare statement:

<b>Parameter</b>	<b>Description</b>	<b>Error Handling</b>
-HIDE	Use to hide the PeopleSoft Application Designer interface. Always use this parameter when performing a command line Compare process.	Required for all project command line processes.
-CT	Source database type. Enter the type of source database to which you are connecting (for example, ORACLE, SYBASE, and so on). Choose from the Database Type Selection table.  See <a href="#">Appendix B, "Referencing Selections for Command Line Parameters," Database Type Selections, page 579.</a>	Required. If the process stops, error messages are written to the log file if the log file name parameter is specified.
-CS	Source server name. Enter the name of the source database server for the database to which you are connecting.	Required for some database types. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-CD	Source database name. Enter the name of the source database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.	Required. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-CO	Source user ID. Enter the PeopleSoft user ID to use to sign in to source database.	Required. If you omit this parameter, the user ID is taken from the registry. If it fails, the process stops and error messages are written to the log file if the log file name parameter is specified.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-CP	Source user password.  Enter the password for the specified user ID for source database.	Required. If you omit this parameter, the PeopleSoft Signon dialog box appears and prompts the user for the password. If the password fails, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TS	Target server name.  Enter the name of the target database server for the database to which you are connecting.	Required for some database types. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TD	Target database name.  Enter the name of the target database to which you are connecting as you would enter it into the PeopleSoft Signon dialog box.	Required. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TO	Target user ID.  Enter the PeopleSoft user ID to use to sign in to the target database.	Required. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-TP	Target user password.  Enter the password for the specified user ID for the target database.	Required. If you omit this parameter, the process stops and error messages are written to the log file if the log file name parameter is specified.
-LF	Log file name.  Enter the name of the file in which error messages are logged during the command line Upgrade Compare process.	Not required. If you omit this parameter, a file is created with the specified path and name, and all of the error messages are written to that file.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-PJM	Source project name.  Enter the name of the project that is to be compared from the source database to the target database. This project should be available in the source database before starting the command line Upgrade Compare process to the target database.	Required. This is the main parameter that is used internally by the executable file to identify an upgrade compare. If you omit this parameter and if you supply all of the source database login parameters, this executable file starts the application.
-CL	Commit limit  Enter the Commit limit (a number greater than 0) for the number of objects to be copied or compared before a commit is issued. Example: -CL 150	Not required. The default is 50.
-AF	Audit flags on records (0 or 1). Enter 1 to keep the target audit flags as is. Enter 0 to take them from the source database. Example: -AF 0	Not required. The default is 1.
-PPL	Portal registry structures permission list references.  Enter 1 to keep the target portal registry permission list references as is. Enter 0 to take them from the source database. Example: -PPL 0	Not required. The default is 1.
-DDL	DDL on records and indexes (0 or 1). Enter 1 to keep the target database DDL flags as is. Enter 0 to take them from the source database. When keeping the target database DDL, no differences are shown on the compare report. Example: -DDL 0	Not required. The default is 1.
-CFD	Chartfield PageField Display Size (0 or 1). Enter 1 to keep the target database chartfield pagefield display size attributes. Enter 0 to take them from the source database. When keeping target database attributes, no differences are shown on the compare report for these attributes. Example -CFD 0	Not required. The default is 1.

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-CFF	Chartfield Field Format (0 or 1). Enter 1 to keep the target database chartfield field format attributes. Enter 0 to take them from the source database. When keeping target attributes, no differences are shown on the compare report for these attributes. Example: -CFF 0	Not required. The default is 1.
-CMT	Compare type: project (1) or database (0). Example: -CMT 0	Not required. The default is 1.
-TGT	Target Orientation: PeopleSoft vanilla (0) or keep personalization (1).	Not required. The default is 0.
-CBY	Comparison By: To compare by release, enter REL, followed by the release number. To compare by date, enter DAT followed by a date in the following format: YYYY-MM-DD-HH.MM.SS.sss. Release numbers are included in the Available Release Numbers list. Example: 2002-02-01-12.00.00.000	Not required. The default is REL.
-ROD	Report Output Directory. Enter the report output directory location. Example-ROD c:\temp\upgreports	Not required. If you omit this parameter, the system uses the Report Output Directory value from the Tools, Options General tab.
-OBJ	Definition type to compare List the definition types to compare as numbers with commas as delimiters, or enter All. Example: -OBJ 0,1  <u>See Appendix B, "Referencing Selections for Command Line Parameters," Definition Type Selections, page 580.</u>	Not required. If you omit this parameter, then all of the definitions are compared by default. For a project comparison, this refers to all definitions in the project. For a database comparison, this includes all definitions in the database. For example, If you choose records and indexes alone for copying, use the appropriate numbers for records and indexes from the Definition Type Selection table. For records, the number is 0, and for the indexes, the number is 1.



<b>Parameter</b>	<b>Description</b>	<b>Error Handling</b>
-LNG	<p>Compare languages.</p> <p>List the language codes with commas as delimiters, or enter ALL to select all available languages. For new languages that are not listed in the Language Selection table, use the xlat short name from the LANGUAGE_CD field in the translate table. For example, if you choose English and Spanish for comparing, enter the appropriate codes for English and Spanish from the Language Selection table. For English, the code is ENG, and for Spanish, the code is ESP. Example: -LNG ENG,ESP</p> <p>See <a href="#">Appendix B, "Referencing Selections for Command Line Parameters," Language Selections, page 586.</a></p>	<p>Not required. If you omit this parameter, then the languages that are already set in the project are used as the default. The COMMON language code refers to nonlanguage attributes that are not translated. Translated attributes can be compared separately from nontranslated attributes. If you do not specify the COMMON code when using -LNG, you compare only the translated language attributes.</p> <p>Changing languages on the command line changes the values that are stored with the project definition. Subsequent Copy and Compare processes on the modified project use the new values unless they are explicitly changed again in the upgrade options.</p>
-CMXML	<p>Generate browser reports (0 or 1).</p> <p>This command enables you to enable the option of generating browser reports written to your report output directory. This action is equivalent to setting the Generate Browser Reports options on the Report Options tab on the Upgrade Options dialog box.</p> <p>Enter 1 to enable this option, and enter 0 to disable this option.</p> <p>Example: -CMXML 1</p> <p>See <i>PeopleTools 8.52: PeopleSoft Application Designer Lifecycle Management Guide</i>, "Comparing Source and Target Definitions," Working with Browser Reports.</p>	<p>Not required. The default is 0.</p>
-CMR [0   1]	<p>Use 0 to disable, 1 to enable, the generation of the Composite Compare Reports. Parameter is only valid when used in conjunction with the -PJM option. Ignored otherwise.</p>	<p>Not required. If not used, then composite report disabled. Must also use -CROD and -CRDB options when using -CMR.</p>

<b><i>Parameter</i></b>	<b><i>Description</i></b>	<b><i>Error Handling</i></b>
-CROD [directory path]	Set path for the Old Release Compare reports when generating Composite Reports (-CMR 1). Only valid when used in conjunction with the -PJM and -CMR 1 options, ignored otherwise.	If not set when generating composite reports (-CMR 1) then its an error condition, and processing should stop, and an error generated to the log file. Directory path can be a local directory, a mapped network directory, or a UNC path specification.
-CRDB [database name]	Pass the value of the Old Release Source Database name when generating Composite Reports. This value must match the Source Database name value from the Old Release Report for each definition type. This is to ensure the original old release report is valid and contains the expected data. If the old release report source database name for the current definition type does not match the passed value, then an error message is generated to the log file, but process continues for other definition types.	Not required.

### ***Example: Running an Upgrade Compare Process***

Use this example as a guide for running an Upgrade Compare process:

- Run a database Compare process against the source Microsoft SQL Server database, PTDMO, and the target database, PTTST.
- Log process and error messages to c:\temp\compare.log.
- Set the commit limit to 150.
- Show the audit flag and DDL differences between databases.
- Compare records, indexes, pages, queries, and process definitions.
- Compare only English and Spanish translations.
- Set the database compare type.
- Set the target orientation to keep the customizations.
- Compare by release 8.40.00.000.
- Generate compare reports to the c:\temp\upgreports directory.

This is a command line statement that performs the preceding list of tasks.

```
PSIDE.EXE -HIDE -CT MICROSOFT -CD PTDMO -CO PTDMO -CP PTDMO -PJM PPLTOOLS -TD  
PTTST -TO PTDMO -TP PTDMO -QUIET -LF C:\TEMP\COMPARE.LOG -CL 150 -AF 0 -DDL 0 -OBJ  
0,1,5,10,20 -LNG ENG, ESP -CMT 0 -TGT 1 -CBY REL Core 8.40.00.000 -ROD C:\TEMP\UPGREPORTS
```



## Chapter 20

# Using Change Control

This chapter provides overviews of Change Control features, implementation considerations, and reporting change control information and discusses how to:

- Set up Change Control.
- Use projects.
- Use Change Control.

---

## Understanding Change Control Features

The Change Control feature has three main functions to help you manage and track development. You can enable one or all of these functions to control how much access each user has to the Change Control commands.

This section discusses:

- Change Control locking
- Change Control history
- Change Control stamping

## Change Control Locking

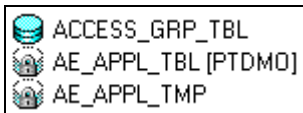
Change Control locking is keyed by PeopleSoft user IDs. When a definition is locked, it cannot be modified by anyone other than the user ID who locked it, and it can be unlocked only by that same user or by a Change Control administrator.

A locked definition is identified by:

- A small padlock icon above the definition in the project workspace (development view) of PeopleSoft Application Designer.
- The user ID of the person who locked the definition, which appears next to the name of the definition.

Your user ID does not appear for definitions that you locked.

For example, the following icons seen in the project workspace display two locked and one unlocked record definitions:



Locked and unlocked record icons

The top record is unlocked; the middle record is locked by another user ID (PTDMO), and the bottom record (with no ID displayed) is locked by you, the current user.

---

**Note.** Standard Change Control locking is supported only for definitions that you can modify with PeopleSoft Application Designer. Other PeopleSoft definitions can be added to a project for upgrading, but they cannot be locked by developers. If you are a Change Control administrator, you can lock all upgradable database definitions, both PeopleSoft Application Designer development definitions and other types. However, for non-PeopleSoft Application Designer definitions, this action prevents unauthorized upgrading only, not unauthorized development.

---

### Locking Projects

With Change Control, projects are treated like all other PeopleSoft Application Designer definitions: you must lock them before you can modify them. However, locking a project does not lock the definitions in the project, and modifying a definition in a project does not modify the project itself.

A project definition includes a name and a list of definitions. When you lock a project definition, other users cannot add or remove definitions from the project, and they cannot rename or delete the project. However, locking does not restrict access to the definitions that are named in the project definition. Likewise, modifying a definition has no affect on the definition of any project to which the definition might belong.

---

**Note.** PeopleSoft Application Designer provides an option to load the last open project on startup. If this option is enabled on your machine and Change Control locking is activated, you might receive an *open in read-only mode?* message at startup if you had not locked the project before or if someone else has the project locked. In either case, you can open the project in read-only mode. Remember, locking does not restrict your access to definitions in the project.

---

### Locking Compared to Version Control

Change control locking is not the same as version control. With a version control system, you check out a copy of a definition and make your changes to the copy. After you check in the changed version, you can always undo your changes. This is not the case with change control locking.

Locking a definition prevents other users from modifying it. However, any changes that you save are written directly to the database, overlaying or replacing the existing definition. You cannot restore a previous version of a definition.

### Locking and Upgrades

When preparing to upgrade a database, all development *must* cease in the source database to assure that the Compare process works with a static environment. It also assures that changes are not made to any definitions between the time that you set the upgrade defaults and the time that you copy the definitions.

You can freeze all development by using the Change Control Administrator dialog box to lock all database definitions. When the upgrade is done, use the same dialog box to unlock all definitions. However, be aware that this action *permanently* removes all previous lock settings from all definitions. Developers have no way of resetting their locks except by manually relocking. When you lock definitions in this way, it is not reflected in the Locked Objects dialog box or in the project workspace. If a developer has unsaved changes when you lock all definitions, the developer cannot save those changes.

For all of these reasons, you must inform developers that you plan to lock all definitions and you must give them time to save their changes, perhaps even to view the Locked Definitions dialog box and print the screen.

Locking definitions with the Change Control Administrator dialog box does not actually mark every definition as locked. Instead, it adds a single row at the top of the locking table. The presence of this row indicates to the system that full database locking is in effect, and it stores the user ID of the administrator who enabled the locking.

Full database locking also plays a role in the target database during the upgrade copy process. During a copy, the system always checks the locking status of the target database definitions to see if they are locked and by whom. If they were locked by a user ID other than the one performing the copy, then those definitions are not modified.

In major upgrades, checking the locking status of each definition before copying severely affects performance. To prevent this, use the Change Control Administrator dialog box to lock all target database definitions before copying. During the copy process, if the entire target database is locked, the system verifies that the user ID performing the copy process is the same user ID that locked the database. If these conditions are true, the system assumes that it was locked for the purpose of the upgrade and that it can safely copy all definitions without checking each one individually.

---

**Note.** When you copy a project, the system does not check the locking status of the definitions in the source database. However, you should keep your definitions locked until the copy is complete.

---

## Change Control History

When Change Control history is enabled, you can enter comments about the modifications that you made to PeopleSoft Application Designer development definitions. History entries contain a common set of information, including who created the entry, when, and the type of action that is associated with the entry. For example, when a user locks a definition, a history entry is automatically created containing the user ID, the date and time, and an action value of Add. This entry can also contain a project name, incident ID, and comment.

### ***Automated History Prompting***

Although you can always insert history entries manually, in many situations PeopleSoft Application Designer automatically inserts history entries and prompts you for comments. In special circumstances, the system adds entries without a prompt. For example, an action history provides some idea of what has happened to a definition, even if no comments were entered. This table describes possible action entries:

<b><i>Action Entry</i></b>	<b><i>Description</i></b>
Lock and Unlock	Whenever a definition is locked or unlocked, you are automatically prompted for a comment, after which a Lock or Unlock action entry is added to the definition history.

<b>Action Entry</b>	<b>Description</b>
Rename	<p>When you rename a definition, you are prompted for a comment, after which a Rename action entry is added to the definition history. The system provides a default comment of &lt;name1&gt; renamed to &lt;name2&gt;:</p> <ul style="list-style-type: none"> <li>• &lt;name1&gt; is the previous name of the definition.</li> <li>• &lt;name2&gt; is the new name that you gave it.</li> </ul> <p><b>Note.</b> If the definition is locked when you try to rename it, the system unlocks it before renaming it and then relocks it afterward. In this case, three history entries are added: one for unlocking, one for renaming, and one for relocking.</p>
Delete	<p>When you delete a PeopleSoft Application Designer definition, its history is retained. During the deletion, PeopleSoft Application Designer prompts you to add a final comment into the definition history, after which a Delete action entry is added to the definition history.</p> <p><b>Note.</b> If Change Control locking is enabled, you can only delete locked definitions. After a locked definition is deleted, it is automatically unlocked and an Unlock action history entry is added. You are not prompted for comments for this unlock event.</p>
Add	<p>When you create a new definition, PeopleSoft Application Designer creates a history entry with an Add action. You are not prompted for comments.</p>
Copy	<p>If Change Control history is enabled in the target database when you perform an upgrade copy, a Copy action entry is inserted into the histories for any added, replaced, or deleted definitions. You are not prompted for comments.</p>

### ***History and Upgrades***

Change Control history is not copied along with its associated definition during an upgrade. However, if history is enabled in the source database, then the history of each affected target definition is updated with a comment noting when the copy was performed and by whom.

This history updating occurs in the background for all target definitions that are affected by the copy process, even non-PeopleSoft Application Designer definitions with histories that you cannot update or view, regardless of whether Change Control history is enabled in the source database.

## **Change Control Stamping**

Change Control stamping is always in effect, regardless of whether locking and history are also enabled. For every definition in the database, PeopleTools maintains a last updated stamp, which denotes the date and time of the last update and the user ID of the person who saved the definition. When a PeopleSoft application delivers a new database, all of the definitions are stamped with a PeopleSoft proprietary ID: PPLSOFT.



## Stamping and Upgrades

Change Control stamping provides critical information during an upgrade comparison. Because the system tracks the user ID of whoever last changed each definition, you can easily identify your adaptations. (Any definition that is stamped with a user ID that is not the PeopleSoft proprietary ID is considered an adaptation.) Whether you made an adaptation before or after the last update is irrelevant; the adaptation is always identified as such.

During a comparison, definitions that you last modified are given a status of Custom Changed (if you have changed them since the compare date) or Custom Unchanged (if you have not changed them since the compare date). PeopleSoft-modified definitions are given a status of either Changed or Unchanged.

---

## Understanding Implementation Considerations

When deciding how to implement Change Control, consider:

- The level of development control that you want to maintain.
- The amount of freedom that developers need.
- The security requirements of the site.

<i>How to Implement</i>	<i>Advantage</i>	<i>Disadvantage</i>
Individual Control	For maximum control of the development environment, use both locking and history, and assign each developer a unique user ID. Then, definitions can be modified by only one developer (user ID) at a time, and developers are always prompted for comments when they lock and unlock definitions.	Developers can share ownership of their definitions only by unlocking them after each change.
Group Control	For flexibility, use locking and history, but assign developers who work on the same project a common ID. Then, developers can share definitions with other members of their group, but not with members of other groups. Administering security is also easier, because you have fewer user IDs to maintain.	The protection from simultaneous development of definitions is decreased. History is harder to track unless developers always include their names in their comments.
History Only	Provides the least restricted Change Control environment. In this situation, all developers can share all definitions.	Developers are not automatically prompted for comments. They can all share the same ID. If you have a very small development team, this option might be good.

## Understanding Change Control Information Reporting

Currently, no predefined reports for retrieving Change Control information are available. However, you can create your own reports by querying the Change Control tables.

The two tables that you can use for reporting are the Change Control History table (PSCHGCTLHIST) and the Change Control Locking table (PSCHGCTLLOCK). These tables have an almost identical column structure:

PSCHGCTLHIST (Record)

Record Fields

Record Type

Num	Field Name	Type	Len	Format	Short Name	Long Name
1	OPRID	Char	30	Mixed	User	User ID
2	OBJECTTYPE	Nbr	5	Raw B	ObjectType	Object Type
3	OBJECTID1	Nbr	5	Raw B	Object Id1	Object Id 1
4	OBJECTVALUE1	Char	100	Mixed	Obj Val 1	Object Value 1
5	OBJECTID2	Nbr	5	Raw B	Object Id2	Object Id 2
6	OBJECTVALUE2	Char	100	Mixed	Obj Val 2	Object Value 2
7	OBJECTID3	Nbr	5	Raw B	Object Id3	Object Id 3
8	OBJECTVALUE3	Char	30	Mixed	Obj Val 3	Object Value 3
9	OBJECTID4	Nbr	5	Raw B	Object Id4	Object Id 4
10	OBJECTVALUE4	Char	30	Mixed	Obj Val 4	Object Value 4
11	DTM_STAMP	DtTm	26	Scnds	Date/Time	Date/Time Stamp
12	CHGCTRL_ACTION	Char	1	Upper	Action	Change Control Action
13	PROJECTNAME	Char	30	Upper	Project	Project Name
14	INCIDENT_ID	Char	18	Mixed	Incident	Incident ID
15	DESCRLONG	Long	0		Descr	Description

Structure of PSCHGCTLHIST

The main difference between these two tables is that PSCHGCTLHIST contains a CHGCTRL\_ACTION field, while PSCHGCTLLOCK does not.

Each PeopleSoft definition in these tables is uniquely identified by numeric codes (DEFINITIONID columns) and names (DEFINITIONVALUE columns). The different DEFINITIONID and DEFINITIONVALUE column pairs correspond to the various definition key types and values for each kind of definition. You can see these definition keys when you view the upgrade definition window. For example, translate values have these keys—Field Name, Field Value, and Effective Date, as in this example:

Translate Values (Upgrade Definition Type)								
Translate Values Key								
	Field Name	Field Value	Effective Date	Source	Target	Action	Upgrade	Done
1	AE_ABEND_ACTION	A	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	AE_ABEND_ACTION	B	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	AE_ABEND_ACTION	I	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	AE_ABEND_ACTION	S	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	AE_ACTIVE_STATUS	A	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	AE_ACTIVE_STATUS	I	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	AE_ADJUST_STATU	A	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	AE_ADJUST_STATU	D	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	AE_ADJUST_STATU	M	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	AE_ADJUST_STATU	X	1900-01-01	Unknown	Unknown	Copy	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Viewing definition keys

In the Change Control tables, the row containing the first translate value in the preceding example has the following field values:

<i>DEFINITIONVALUE1</i>	<i>DEFINITIONVALUE2</i>	<i>DEFINITIONVALUE3</i>
AE_ABEND_ACTION	A	1900-01-01

When reporting on a particular definition type, you want to retrieve definition values, but you must limit the query by using the definition IDs for the definition type. The following tables list all of the upgradable definition types, their corresponding definition ID codes, and the type of value that each ID represents (in parentheses).

## PeopleTools Definition Types

This table lists PeopleTools definition types and their definition IDs:

<i>Definition Type</i>	<i>DEFINITIONID1</i>	<i>DEFINITIONID2</i>	<i>DEFINITIONID3</i>	<i>DEFINITIONID4</i>
Access Groups	17 (name)	0	0	0
Activities	18 (name)	0	0	0
Application Engine Programs	66 (name)	Not applicable (NA)	NA	NA
Application Engine Sections	66 (name)	77 (section)	NA	NA
Application Package	104 (name)	NA	NA	NA

<b>Definition Type</b>	<b>DEFINITIONID1</b>	<b>DEFINITIONID2</b>	<b>DEFINITIONID3</b>	<b>DEFINITIONID4</b>
Approval Rule Sets	85 (name)	21 (effective date)	NA	NA
Business Interlink	64 (name)	NA	NA	NA
Business Processes	7 (name)	NA	NA	NA
Colors	19 (name)	25 (user ID)	0	0
Components	10 (name)	39 (market)	NA	NA
Component Interfaces	74 (name)	NA	NA	NA
Cube Definitions	54 (name)	55 (description)	NA	NA
Cube Instance Definitions	56 (name)	57 (description)	NA	NA
Dimensions	51 (name)	52 (dimension type)	53 (description)	0
Fields	6 (name)	0	0	0
Field Formats	23 (family name)	0	0	0
File Layout Definitions	71 (name)	NA	NA	NA
HTML	90 (name)	95 (type)	NA	NA
Images	91 (name)	95 (type)	NA	NA
Indexes	1 (name)	24 (index ID)	0	0
Job Definitions	27 (name)	NA	NA	NA
Menus	3 (name)	0	0	0

<i>Definition Type</i>	<i>DEFINITIONID1</i>	<i>DEFINITIONID2</i>	<i>DEFINITIONID3</i>	<i>DEFINITIONID4</i>
Message Catalog Entries	48 (message set number)	48 (message number)	16 (language code)	50 (description)
Message Channels	61 (name)	NA	NA	NA
Message Definitions	60 (name)	NA	NA	NA
Mobile Page <b>Important!</b> PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.	111 (name)	NA	NA	NA
Message Nodes	62 (name)	NA	NA	NA
Pages	9 (name)	NA	0	0
PeopleCode	See PeopleCode Definition Types	NA	NA	NA
Problem Type	109 (name)	NA	NA	NA
Process Definitions	29 (process type)	28 (name)	0	0
Process Type Definitions	29 (name)	26 (operating system)	20 (database type)	NA
Queries	30 (name)	25 (user ID)	0	0
Records	1 (name)	2 (RecField name)	0	0
Recurrence Definitions	31 (name)	NA	NA	NA
Roles	32 (name)	0	0	0

<i>Definition Type</i>	<i>DEFINITIONID1</i>	<i>DEFINITIONID2</i>	<i>DEFINITIONID3</i>	<i>DEFINITIONID4</i>
Server Definitions	33 (name)	NA	NA	NA
SQL	65 (name)	81 (SQL type)	NA	NA
Styles	35 (name)	0	0	0
Style Sheets	94 (name)	NA	NA	NA
Translate Values	6 (database field name)	22 (value)	21 (effective date)	NA
Trees	34 (setID)	68 (user key value)	36 (tree name)	21 (effective date)
Tree Structures	37 (name)	0	0	0

## PeopleCode Definition Types

This table lists PeopleCode definition types and their definition IDs:

<i>Definition Type</i>	<i>DEFINITIONID1</i>	<i>DEFINITIONID2</i>	<i>DEFINITIONID3</i>	<i>DEFINITIONID4</i>
Application Engine	66 (PeopleSoft Application Engine program)	77 (section, market, database type, effective date)	78 (step)	12 (method)
Application Package	104 (application package)	NA	NA	NA
Component Interface	74 (business component)	12 (method)	NA	NA
Menu	3 (menu)	4 (bar)	5 (item)	12 (method)
Message	60 (message)	12 (method)	NA	NA
Page	9 (panel)	16 (language code)	12 (method)	NA
Page Field	9 (panel)	16 (language code)	67 (field)	12 (method)

<i>Definition Type</i>	<i>DEFINITIONID1</i>	<i>DEFINITIONID2</i>	<i>DEFINITIONID3</i>	<i>DEFINITIONID4</i>
Component	10 (panel group)	39 (market)	12 (method)	NA
Component Record	10 (panel group)	39 (market)	1 (record)	12 (method)
Component Record Field	10 (panel group)	39 (market)	1 (record)	2 (fieldname, method)
Record	1 (record)	2 (field)	12 (method)	NA
Subscription	60 (message)	87 (subscription)	12 (method)	NA

## Change Control Supported Definition Types

These are the supported definition types for change control:

- Activity
- Application Engine Program
- Application Package
- Approval Rule Set
- Business Interlink
- Business Process
- Component
- Component Interface
- Field
- File Layout
- HTML
- Image
- Menu
- Message
- Message Channel
- Mobile Page
- Problem Type

- Page
- Project
- Record
- SQL
- Style Sheet

When reporting on Change Control history, consider one other field: `CHGCTRL_ACTION`. This field stores the one-letter code for the various actions that Change Control history tracks:

- A (Add)
- C (Copy)
- D (Delete)
- I (Informational)
- L (Lock)
- R (Rename)
- U (Unlock)

Here's an example of a SQL query to report on all deleted definitions:

```
select oprid, definitionvalue1, definitionvalue2,
definitionvalue3, definitionvalue4, dtm-_stamp, projectname,
incident_id, descrlong
from pschgctlhist
  where chgctrl_action = 'D'
  order by oprid, definitionvalue1
```

---

**Note.** Full history tracking is supported only for PeopleSoft Application Designer definitions: business processes, business process maps, fields, menus, panels, panel groups, projects, and records. Other definition types have history entries only when the `CHGCTL_ACTION` field value is `C`, and only if they have been upgraded.

---

## Setting Up Change Control

This section provides an overview of Change Control security and discusses how to:

- Appoint a Change Control administrator.
- Enable or disable Change Control.

## Understanding Change Control Security

Using the Security component, you can assign users one of three Change Control access levels, depending on how much authority you want them to have. This table describes the available access levels:



<b>Change Control Access Level</b>	<b>Description</b>
Restricted access	Restricts users from locking or unlocking definitions. When Change Control locking is enabled, users with restricted access can open PeopleSoft Application Designer development definitions in read-only mode only. Users are also unable to view or update definition histories.
Developer access	Allows users to lock any unlocked definitions and to unlock any definitions that they have locked. They can then manipulate definitions as they are allowed to in their security profile. Users can also view and enter definition history comments.
Supervisor access	Allows users to unlock any locked definitions, regardless of who locked them. They can also access the Change Control Administrator dialog box, in which you lock and unlock all definitions at one time and enable and disable Change Control locking and history.

If Change Control locking is disabled, these access levels have no security value. If history is disabled, users with developer and supervisor access can still view the History dialog box, but users with restricted access cannot.

Remember that Change Control is based on user ID. If developers all share the same user ID, then Change Control offers no advantage in control because each developer can modify definitions that are locked by others.

## Appointing a Change Control Administrator

Appoint Change Control administrators by giving certain users supervisor-level access to Change Control. When users have this access level, they can enable and disable Change Control.

See *PeopleTools 8.52: Security Administration*, "Setting Up Permission Lists," Setting PeopleTools Permissions.

## Enabling or Disabling Change Control

To enable or disable Change Control:

1. In PeopleSoft Application Designer, select Tools, Change Control, Administrator.

The Change Control Administrator dialog box appears.

## 2. Specify the System Wide Options and access control:

<b>Use change control locking</b>	Select to require that PeopleSoft Application Designer definitions be locked before they can be modified.
<b>Use change control history</b>	<p>Select to enable PeopleSoft Application Designer developers to insert comments about open definitions.</p> <p>If both Use change control locking and Use change control history check boxes are selected, developers are prompted for comments when locking and unlocking definitions.</p> <hr/> <p><b>Note.</b> Because these are system-wide settings, if you change them all users must log off and on again for the changes to take effect.</p> <hr/>
<b>Lock all definitions</b>	<p>Select or deselect to lock or unlock all definitions in the database.</p> <p>Usually, you lock all definitions only before a major upgrade, because it permanently removes all individual developer locks.</p>

---

## Using Projects

This section provides an overview of projects and discusses how to:

- Use multiple databases for development.
- Use distributed development environments.

## Understanding Projects

You can use all levels of Change Control with or without also using projects. If you decide not to use projects, you rely on the Locked Objects dialog box, rather than the project workspace, to identify locked definitions. The dialog box provides a better overall view of locking status because it shows all of the PeopleSoft Application Designer definitions in the database, not just those in the current project.

Use projects to track the definitions that are changed as part of a change or feature request. This set of definitions is commonly referred to as the change set. The Options dialog box has an option to insert a definition into a project when it is modified and saved. If you start with an empty project, this option provides an easy way of tracking the change set for this incident. When the change request is completed, the project contains everything that is associated with the change. You should also use the Comments field in the Project Properties dialog box to list any external definitions, like COBOL or Structured Query Report (SQR) modules, that must be migrated with this change.

## Using Multiple Databases for Development

Managing change in a single database environment is straightforward, but very few PeopleSoft users operate in a single database environment. The classic development model uses three databases: development, test, and production. You apply all changes to the development database. After you unit-test the change, migrate the change set to the test database, where it goes through more rigorous testing. Usually, you would run one or more regression test suites to ensure that you resolve the issue that you intended to resolve with no unwelcome side effects. Finally, that change set is migrated into the production database. If you find a problem at any stage in the process, the incident is sent back to development and the process begins again.

This model assumes that the development database is the master database. You can use the Change Control locking feature to lock down the modules on which you and other developers are working. When developers complete the changes in the development database, they notify the Change Control administrator and use the upgrade copy facility to copy the change set into the test environment. As long as you use the technique that is described previously in this section, the project should contain the entire change set. The system tracks all of the documentation for the change in the development database. The only information that appears in the test and production databases is a history line indicating it was copied. Definitions move only in one direction in this model: from development to test, then from test to production.

---

**Note.** The only case in which you can copy a definition back to development from either test or production is if a problem must be recreated and another change was already made to the affected definition. Do this with extreme care because upgrade copies are destructive and cannot be undone if you discover that you overlaid another developer's change. For this reason, you should apply changes directly to test or production databases only rarely.

---

---

**Note.** This Change Control model is just one that you can use. We provide it to give you an idea of how you can implement Change Control in your environment. While you do not need to follow this model exactly, you should implement a Change Control model that enables you to track changes to the system and prevent developers from overwriting each other's changes.

---

## Using Distributed Development Environments

You should use a master development database even if each development team works on its own copy of the database. Developers should lock down the definitions on which they intend to work in the master development database, and then copy those modules to their private databases. This method ensures that no other developer makes a change to those definitions while they are checked out.

When a developer is ready to copy changes back to the master development database, you check the Change Control history of the locked definitions in the master development database. Do this before using upgrade copy to migrate them back, just in case a Change Control administrator has overridden a lock and made a change while the definitions were checked out.

---

**Note.** Change Control administrators should always notify the developer who has a lock on a definition before they override to avoid unexpected surprises later.

---

---

## Using Change Control

This section discusses how to:

- Lock and unlock definitions.
- View locked definitions.
- Insert comments.
- Delete page definitions.
- View Change Control history.

## Locking and Unlocking Definitions

You can lock and unlock definitions manually. You can also have PeopleSoft Application Designer lock unlocked definitions for you each time that you open them.

You must have developer or supervisor access to Change Control to lock and unlock definitions. If you have supervisor access, you can also lock all definitions at once, which can be helpful when performing upgrades to ensure that definitions are not modified in the middle of the process.

This section discusses how to:

- Lock or unlock an unopened definition in the current project.
- Unlock an unopened definition that is not in the current project.
- Lock or unlock an open definition.
- Lock definitions automatically when you open them.
- Lock or unlock all definitions at once.

### ***Locking or Unlocking an Unopened Definition in the Current Project***

To lock or unlock an unopened definition in the current project:

1. In the project workspace, select the Development tab at the base of the workspace window to activate the development view.
2. Select the definitions that you want to lock or unlock.
3. Right-click any of the selected definitions and select Lock Definition or Unlock Definition.

### ***Unlocking an Unopened Definition Not in the Current Project***

To unlock an unopened definition that is not in the current project:

1. Select Tools, Change Control, View Locked Definitions.

The Locked Definitions dialog box appears.

---

**Note.** The View Locked Definitions menu item is not available if you have restricted access to Change Control.

---

2. Select the definition type and user.

You can view all locked definitions of the specified type by selecting *all* from the User drop-down list box.

3. Select the definitions to unlock.
4. Right-click any selected definition and select Unlock Definition.

If Change Control history is enabled, the system will prompt you for comments.

### ***Locking or Unlocking an Open Definition***

To lock or unlock an open definition:

1. Activate the definition.
2. Select Tools, Change Control, and then select either Lock Definition or Unlock Definition.

### ***Locking Definitions Automatically When You Open Them***

To lock definitions automatically when you open them:

1. Select Tools, Options, Change Control.
2. Select Lock definition when it is opened.

Now, whenever you open a definition it is locked automatically, unless you have only restricted access to Change Control. In this case, the system will notify you that you have restricted access and will ask whether you want to open the definition in read-only mode.

---

**Note.** Like all settings in the Application Designer Options dialog box, this setting controls the behavior on your workstation only. Also, definitions cannot be unlocked automatically; you must always unlock them manually.

---

### ***Locking or Unlocking All Definitions at Once***

To lock or unlock all definitions at once:

1. Select Tools, Change Control, Administrator.

The Change Control Administrator dialog box appears.

2. Select Lock all definitions to lock all definitions or deselect it to unlock all definitions.

Locking all definitions applies a database-wide lock that is tagged with your user ID.

---

**Warning!** Selecting this check box removes individual locks from all database definitions. You should proceed with this step only if you have informed all of your developers and given them an opportunity to save any unsaved work.

---

3. Click OK.

If you are locking all definitions, the system warns you that this action permanently cancels existing locks. Click Yes.

## Viewing Locked Definitions

At times you may want to see a definition even if it is locked.

To view locked definitions:

1. Select Tools, Change Control, View Locked Definitions.

The Locked Definitions dialog box appears.

2. Select the user whose locked definitions you want to view.
3. Select the type to display.

You can view only one definition type at a time. You can also unlock definitions in this dialog box.

## Inserting Comments

When Change Control history is enabled, you can insert comments about an open definition at any time. To help ensure that you insert new comments with each modification, you can instruct PeopleSoft Application Designer to prompt you for a comment every time you save a definition, every time you lock or unlock a definition, or both.

This section discusses how to:

- Insert a comment for an unopened definition in the current project.
- Insert a comment for an open definition.
- Enable a prompt for comments when saving a definition.

### ***Inserting a Comment for an Unopened Definition in the Current Project***

To insert a comment for an unopened definition in the current project:

1. In the project workspace, select the Development tab at the base of the workspace window to activate the development view.

2. Select the definition for which you want to insert comments.

You cannot insert comments for more than one definition at a time.

3. Right-click the selected definition and select Insert Comment.

The Insert Comment dialog box appears.

4. Enter the necessary information:

<b>Project</b>	Displays the name of the current project by default, but you can delete or replace this value.
<b>Incident ID</b>	Enter the incident identification to which your development corresponds.
<b>Comments</b>	Enter information about why and how you are modifying the definition.

If you click OK, the information is inserted into the definition history and the dialog box closes. If you click Apply, your comments are inserted, but the dialog box remains open. You can then enter comments for another history entry. When you click OK or Apply, the new comments are inserted as a new history entry; they do not replace the previous entry.

### ***Inserting a Comment for an Open Definition***

To insert a comment for an open definition:

1. Use the list in the Window menu to navigate to the definition.
2. Select Tools, Change Control, Insert Comment.
3. Enter the name of the project and incident ID.
4. Enter your comments.
5. Click OK or Apply.

### ***Enabling a Prompt for Comments When Saving a Definition***

To be prompted for comments when saving a definition:

1. Select Tools, Options, Change Control.

The Change Control page appears.

2. Select Prompt for comments when definition is saved.

Whenever you save a definition, the system prompts you to insert history comments.

---

**Note.** This setting affects the behavior on your workstation only. One possible drawback to using this setting is that a definition might be saved many times as part of a single change, and you will be prompted for comments at every save.

---

## Deleting Page Definitions

When deleting page definitions with Change Control enabled, the system presents this message:

Are you sure you want to delete <definition name> definition? You are presented with Yes and No options. If you click the Yes button and the page is locked, then the system will delete the page. If you click the Yes button and the page is not locked, then this message box appears:

You must lock <definition name> before you may rename or delete it.

You must lock the page before you are allowed to delete it.

If you click the No button, then the system does not delete the definition.

---

**Note.** If the definition is a base language page, then the system also deletes all non-base language pages. If the definition is a non-base language page, then the system deletes only the selected non-base language page.

---

If you select multiple pages to delete but the base language page is not included in the selection, then the system prompts you for lock comments for each selected non-base language page. If you select multiple pages to delete and the base language page is included in the selection, then the system prompts you for lock comments for only the base language page.

### See Also

*PeopleTools 8.52: Global Technology*, "Developing Global Applications," Designing Global-Ready Pages

## Viewing Change Control History

For every PeopleSoft Application Designer development definition in the database, you can view its Change Control history.

To view Change Control history:

1. Select Tools, Change Control, View History.

The History dialog box appears.



## 2. Select a definition type and name.

The Definition Name list box contains only the names of PeopleSoft Application Designer definitions that have at least one history entry. Click the Refresh button to ensure that you are viewing the most recent listing of locked definitions.

The history table contains these columns:

<b>Date</b>	Displays when each entry was added.
<b>User</b>	Displays who added an entry.
<b>Action</b>	Displays why a comment was entered. Five of these Action types—Lock, Informational, Unlock, Rename, and Delete—represent actions that you perform in PeopleSoft Application Designer and for which the system prompts for comments.
<b>Add</b>	Denotes an automatic history entry that PeopleTools inserts when a new definition is created. In this case, and whenever the system performs a background lock or unlock, the system does not prompt you for comments. The Comment column contains the text System Generated.
<hr/>	
	<b>Note.</b> PeopleSoft Application Designer performs automatic locks and unlocks under certain situations. For example, when you rename a locked definition, that definition must be unlocked before it is renamed and then relocked afterward. Likewise, when you delete a locked definition, the definition is automatically unlocked following the deletion. The system does not prompt you for comments during any of these unlock or relock actions, but it adds corresponding history entries automatically.
<hr/>	
<b>Copy</b>	The system adds this type of history entry automatically when a definition is copied into the current database. In this case, the Project value is the name of the copied project in the source database, and no comments are added.
<b>Project, Incident ID, and Comment</b>	Displays the project name, incident ID, and relevant comments.

## 3. Double-click a row in the grid to open a history entry, if necessary.

The History Details dialog box appears. This dialog box is a read-only version of the Insert Comments dialog box. You cannot update the information that appears in this dialog box.



## Appendix A

# Understanding Functional Indexes

This appendix discusses functional indexes for:

- Microsoft SQL Server 2000.
- DB2/UDB for Linux, UNIX, and Windows.
- Informix.

---

## Functional Indexes for Microsoft SQL Server 2000

The following example shows how the key fields are concatenated:

```

USE FS840U70
go
SET IMPLICIT_TRANSACTIONS ON
go
IF EXISTS (SELECT 'X' FROM SYSOBJECTS WHERE TYPE = 'U' AND NAME =
'PS_QE_ALTER_NEW') DROP TABLE PS_QE_ALTER_NEW
go
CREATE TABLE PS_QE_ALTER_NEW (QE_ALTER_FLD1 NCHAR(1) NOT NULL,
    QE_ALTER_FLD2 NCHAR(1) NOT NULL,
    QE_ALTER_FLD3 NCHAR(1) NOT NULL,
    QE_ALTER_FLD4 NCHAR(10) NOT NULL,
    QE_ALTER_FLD5 NCHAR(20) NOT NULL,
    QE_ALTER_FLD6 NCHAR(11) NOT NULL,
    QE_ALTER_FLD7 NCHAR(1) NOT NULL,
    QE_ALTER_FLD8 NCHAR(1) NOT NULL,
    QEPC_AGE SMALLINT NOT NULL,
    QEPC_FILE_NUM DECIMAL(6, 2) NOT NULL,
    QE_BEGIN_DT PSDATE NULL,
    QE_BIRTHDATE PSDATE NULL,
    QE_31DIGFLD9 DECIMAL(31, 8) NOT NULL,
    QE_ANNUAL_PCT DECIMAL(3, 1) NOT NULL,
    QE_MS_DTTM1 PSDATETIME NULL,
    QE_MS_DTTM2 PSDATETIME NULL,
    QE_ALTER_TIMEA PTIME NULL,
    QE_ALTER_TIMEB PTIME NULL,
    MSSCONCATCOL AS QE_ALTER_FLD1 + QE_ALTER_FLD2 + QE_ALTER_FLD3 +
QE_ALTER_FLD4 + QE_ALTER_FLD5 + QE_ALTER_FLD6 + QE_ALTER_FLD7 +
QE_ALTER_FLD8 + CONVERT(NCHAR,QEPC_AGE) + CONVERT(NCHAR
,QEPC_FILE_NUM) + CONVERT(NCHAR(16),QE_BEGIN_DT, 109) +
CONVERT(NCHAR(16),QE_BIRTHDATE, 109) + CONVERT(NCHAR,QE_31DIGFLD9) +
CONVERT(NCHAR,QE_ANNUAL_PCT) + CONVERT(NCHAR(16),QE_MS_DTTM1, 109) +
CONVERT(NCHAR(16),QE_MS_DTTM2, 109) + CONVERT(NCHAR(16)
,QE_ALTER_TIMEA, 109))
go
COMMIT
go
SET ARITHABORT ON
go
CREATE UNIQUE CLUSTERED INDEX PS_QE_ALTER_NEW ON PS_QE_ALTER_NEW
(MSSCONCATCOL)
go
CREATE NONCLUSTERED INDEX PSWQE_ALTER_NEW ON PS_QE_ALTER_NEW
(QE_ALTER_FLD1,
    QE_ALTER_FLD2,
    QE_ALTER_FLD3,
    QE_ALTER_FLD4,
    QE_ALTER_FLD5,
    QE_ALTER_FLD6,
    QE_ALTER_FLD7,
    QE_ALTER_FLD8,
    QEPC_AGE,
    QEPC_FILE_NUM,
    QE_BEGIN_DT,
    QE_BIRTHDATE,
    QE_31DIGFLD9,
    QE_ANNUAL_PCT,
    QE_MS_DTTM1,
    QE_MS_DTTM2)
go
COMMIT
Go

```

## Functional Indexes for DB2/UDB for Linux, UNIX, and Windows

The following example shows how the key fields are concatenated:

```
CREATE TABLE PS_QE_ALTER_NEW (QE_ALTER_FLD1 VARGRAPHIC(1) NOT NULL,
    QE_ALTER_FLD2 VARGRAPHIC(1) NOT NULL,
    QE_ALTER_FLD3 VARGRAPHIC(1) NOT NULL,
    QE_ALTER_FLD4 VARGRAPHIC(10) NOT NULL,
    QE_ALTER_FLD5 VARGRAPHIC(20) NOT NULL,
    QE_ALTER_FLD6 VARGRAPHIC(11) NOT NULL,
    QE_ALTER_FLD7 VARGRAPHIC(1) NOT NULL,
    QE_ALTER_FLD8 VARGRAPHIC(1) NOT NULL,
    QEPC_AGE VARGRAPHIC(8) NOT NULL,
    QEPC_FILE_NUM DECIMAL(6, 2) NOT NULL,
    QE_BEGIN_DT DATE,
    QE_BIRTHDATE DATE,
    QE_31DIGFLD9 DECIMAL(31, 8) NOT NULL,
    QE_ANNUAL_PCT DECIMAL(3, 1) NOT NULL,
    QE_MS_DTTM1 TIMESTAMP,
    QE_MS_DTTM2 TIMESTAMP,
    QE_ALTER_TIMEA TIME,
    QE_ALTER_TIMEB TIME,
    DBXCONCATCOL GENERATED ALWAYS AS (QE_ALTER_FLD1 || QE_ALTER_FLD2 ||
QE_ALTER_FLD3 || QE_ALTER_FLD4 || QE_ALTER_FLD5 || QE_ALTER_FLD6 ||
QE_ALTER_FLD7 || QE_ALTER_FLD8 || QEPC_AGE ||
VARGRAPHIC(CHAR(QEPC_FILE_NUM)) ||
VALUE(VARGRAPHIC(CHAR(QE_BEGIN_DT)), '')) ||
VALUE(VARGRAPHIC(CHAR(QE_BIRTHDATE)), '')) ||
VARGRAPHIC(CHAR(QE_31DIGFLD9)) || VARGRAPHIC(CHAR(QE_ANNUAL_PCT)) ||
VALUE(VARGRAPHIC(CHAR(QE_MS_DTTM1)), '')) ||
VALUE(VARGRAPHIC(CHAR(QE_MS_DTTM2)), '')) ||
VALUE(VARGRAPHIC(CHAR(QE_ALTER_TIMEA)), ''))) IN PTAPP INDEX IN
PTAPPIDX NOT LOGGED INITIALLY;
COMMIT;
CREATE UNIQUE INDEX PS_QE_ALTER_NEW ON PS_QE_ALTER_NEW (DBXCONCATCOL);
CREATE INDEX PSWQE_ALTER_NEW ON PS_QE_ALTER_NEW (QE_ALTER_FLD1,
    QE_ALTER_FLD2,
    QE_ALTER_FLD3,
    QE_ALTER_FLD4,
    QE_ALTER_FLD5,
    QE_ALTER_FLD6,
    QE_ALTER_FLD7,
    QE_ALTER_FLD8,
    QEPC_AGE,
    QEPC_FILE_NUM,
    QE_BEGIN_DT,
    QE_BIRTHDATE,
    QE_31DIGFLD9,
    QE_ANNUAL_PCT,
    QE_MS_DTTM1,
    QE_MS_DTTM2);
COMMIT;
```

## Functional Indexes for Informix

The following example shows how the key fields are concatenated:

```

BEGIN WORK;
CREATE TABLE PS_QE_ALTER_NEW (QE_ALTER_FLD1 CHAR(1) NOT NULL,
    QE_ALTER_FLD2 CHAR(1) NOT NULL,
    QE_ALTER_FLD3 CHAR(1) NOT NULL,
    QE_ALTER_FLD4 CHAR(10) NOT NULL,
    QE_ALTER_FLD5 CHAR(20) NOT NULL,
    QE_ALTER_FLD6 CHAR(11) NOT NULL,
    QE_ALTER_FLD7 CHAR(1) NOT NULL,
    QE_ALTER_FLD8 CHAR(1) NOT NULL,
    QEPC_AGE SMALLINT NOT NULL,
    QEPC_FILE_NUM DECIMAL(6, 2) NOT NULL,
    QE_BEGIN_DT DATE,
    QE_BIRTHDATE DATE,
    QE_31DIGFLD9 DECIMAL(31, 8) NOT NULL,
    QE_ANNUAL_PCT DECIMAL(3, 1) NOT NULL,
    QE_MS_DTTM1 DATETIME YEAR TO FRACTION(3),
    QE_MS_DTTM2 DATETIME YEAR TO FRACTION(3),
    QE_ALTER_TIMEA DATETIME HOUR TO FRACTION(3),
    QE_ALTER_TIMEB DATETIME HOUR TO FRACTION(3)) IN PTAPP LOCK MODE ROW;
REVOKE ALL ON PS_QE_ALTER_NEW FROM PUBLIC;
COMMIT;
BEGIN WORK;
CREATE FUNCTION PS_QE_ALTER_NEW_SP1(
    QE_ALTER_FLD1 CHAR(1),
    QE_ALTER_FLD2 CHAR(1),
    QE_ALTER_FLD3 CHAR(1),
    QE_ALTER_FLD4 CHAR(10),
    QE_ALTER_FLD5 CHAR(20),
    QE_ALTER_FLD6 CHAR(11),
    QE_ALTER_FLD7 CHAR(1),
    QE_ALTER_FLD8 CHAR(1),
    QEPC_AGE SMALLINT,
    QEPC_FILE_NUM DECIMAL,
    QE_BEGIN_DT DATE,
    QE_BIRTHDATE DATE,
    QE_31DIGFLD9 DECIMAL,
    QE_ANNUAL_PCT DECIMAL,
    QE_MS_DTTM1 DATETIME YEAR TO FRACTION(3),
    QE_MS_DTTM2 DATETIME YEAR TO FRACTION(3))
    RETURNING VARCHAR(166) WITH (NOT VARIANT);
RETURN QE_ALTER_FLD1||
    QE_ALTER_FLD2||
    QE_ALTER_FLD3||
    QE_ALTER_FLD4||
    QE_ALTER_FLD5||
    QE_ALTER_FLD6||
    QE_ALTER_FLD7||
    QE_ALTER_FLD8||
    QEPC_AGE||
    QEPC_FILE_NUM||
    QE_BEGIN_DT||
    QE_BIRTHDATE||
    QE_31DIGFLD9||
    QE_ANNUAL_PCT||
    QE_MS_DTTM1||
    QE_MS_DTTM2;
END FUNCTION;
CREATE FUNCTION PS_QE_ALTER_NEW_SP2(
    QE_ALTER_TIMEA DATETIME HOUR TO FRACTION(3))
    RETURNING VARCHAR(15) WITH (NOT VARIANT);
RETURN QE_ALTER_TIMEA;
END FUNCTION;
CREATE UNIQUE INDEX PS_QE_ALTER_NEW ON PS_QE_ALTER_NEW
    (PS_QE_ALTER_NEW_SP1(

```

```

QE_ALTER_FLD1,
QE_ALTER_FLD2,
QE_ALTER_FLD3,
QE_ALTER_FLD4,
QE_ALTER_FLD5,
QE_ALTER_FLD6,
QE_ALTER_FLD7,
QE_ALTER_FLD8,
QEPC_AGE,
QEPC_FILE_NUM,
QE_BEGIN_DT,
QE_BIRTHDATE,
QE_31DIGFLD9,
QE_ANNUAL_PCT,
QE_MS_DTTM1,
QE_MS_DTTM2),
PS_QE_ALTER_NEW_SP2(
QE_ALTER_TIMEA)) IN FS840IDX;
CREATE INDEX PSWQE_ALTER_NEW ON PS_QE_ALTER_NEW (QE_ALTER_FLD1,
QE_ALTER_FLD2,
QE_ALTER_FLD3,
QE_ALTER_FLD4,
QE_ALTER_FLD5,
QE_ALTER_FLD6,
QE_ALTER_FLD7,
QE_ALTER_FLD8,
QEPC_AGE,
QEPC_FILE_NUM,
QE_BEGIN_DT,
QE_BIRTHDATE,
QE_31DIGFLD9,
QE_ANNUAL_PCT,
QE_MS_DTTM1,
QE_MS_DTTM2) IN FS840IDX;
COMMIT;

```



## Appendix B

# Referencing Selections for Command Line Parameters

This appendix discusses values that are useful for specifying command line parameters; these values are included in these tables:

- Database type selections
- Definition type selections
- Language selections
- Available release numbers

---

## Database Type Selections

This table lists the database type values:

<b><i>Database</i></b>	<b><i>Database Description</i></b>
DB2ODBC	DB2 UDB for OS/390 and z/OS
DB2UNIX	DB2 UDB for Linux, Unix, and Windows
INFORMIX	Informix
MICROSFT	Microsoft SQL Server
ORACLE	Oracle
SYBASE	Sybase

## Definition Type Selections

This table lists the definition type values:

<i>Number</i>	<i>Definition Description</i>
0	Records
1	Indexes
2	Fields
3	Format definitions
4	Translate values
5	Pages
6	Menus
7	Components
8	Record PeopleCode
9	Menu PeopleCode
10	Queries
11	Tree structures
12	Trees*
13	Access groups*
14	Colors
15	Styles

<b><i>Number</i></b>	<b><i>Definition Description</i></b>
16	Not used
17	Business processes
18	Activities
19	Roles*
20	Process definitions
21	Process servers
22	Process types
23	Process jobs
24	Process recurrences
25	Message catalog entries*
26	Dimensions*
27	Cube definitions*
28	Cube instance definitions*
29	Business interlinks
30	SQL definitions
31	File layout definitions
32	Component interfaces
33	Application Engine programs

<b><i>Number</i></b>	<b><i>Definition Description</i></b>
34	Application Engine sections
35	Message nodes
36	Message channels
37	Message definitions
38	Approval rule sets
39	Message PeopleCode
40	Subscription PeopleCode
41	Not used
42	Component interface PeopleCode
43	Application engine PeopleCode
44	Page PeopleCode
45	Page field PeopleCode
46	Component PeopleCode
47	Component record PeopleCode
48	Component record field PeopleCode
49	Images
50	Style sheets
51	HTML

<i>Number</i>	<i>Definition Description</i>
52	Not used
53	Permission lists
54	Portal registry definitions
55	Portal registry structures
56	URL definitions
57	Application Packages
58	Application Package PeopleCode
59	Portal Registry User homepages
60	Problem type definitions
61	Archive templates (deprecated in PeopleTools release 8.44 and above)
62	XSLT
63	Portal Registry User Favorites
64	<p>Mobile pages</p> <p><b>Important!</b> PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.</p>
65	Relationships
66	Component Interface Property PeopleCode**
67	Optimization Models***
68	File References***

<i><b>Number</b></i>	<i><b>Definition Description</b></i>
69	File Reference Type Code***
70	Archive object definitions***
71	Archive Templates (Type 2)***
72	Diagnostic Plug-Ins***
73	Analytic Models
74	Not Used
75	Java Portlet User Preferences
76	WSRP Remote Producers
77	WSRP Remote Portlets
78	WSRP Cloned Portlet Handles
79	Services
80	Service Operations
81	Service Operation Handlers
82	Service Operation Version
83	Service Operation Routings
84	IB Queues
85	BIP Template Definitions
86	BIP Report Definitions

<b><i>Number</i></b>	<b><i>Definition Description</i></b>
87	BIP File Definitions
88	BIP Data Source Definitions
89	WSDL
90	Message Schemas
91	Connected Query Definition
92	Logical Schemas
93	XML Schemas
94	Relational Schemas
95	Dependency Documents
96	Document Schema
97	Essbase Cube Dimensions
98	Essbase Cube Outlines
99	Essbase Cube Connections
100	Essbase Cube Templates
101	Delimited Schemas
102	Positional Schemas
103	Application Data Set Definitions
104	Test Definitions

<b><i>Number</i></b>	<b><i>Definition Description</i></b>
105	Test Framework Test Cases
106	Application Data Set Bindings
107	Feed Definitions
108	Feed Categories
109	Feed Data Type
110	JSON Documents
111	Related Content Definition
112	Related Content Services
113	Related Content Configurations
114	Related Content Layouts
115	Search Attributes
116	Search Definitions
117	Search Categories
118	Search Contexts
119	Integration Groups

---

## Language Selections

This table lists the language code values:



<b>Code</b>	<b><i>Language Description</i></b>
ARA	Arabic
BUL	Bulgarian
CFR	Canadian French
CRO	Croatian
CZE	Czech
DAN	Danish
DUT	Dutch
ENG	English
ESP	Spanish
FIN	Finnish
FRA	French
GER	German
GRK	Greek
HEB	Hebrew
HUN	Hungarian
ITA	Italian
JPN	Japanese
KOR	Korean

<b>Code</b>	<b><i>Language Description</i></b>
MAY	Bahasa Malay
NOR	Norwegian
POL	Polish
POR	Portuguese
ROM	Romanian
RUS	Russian
SER	Serbian
SLK	Slovak
SLV	Slovenian
SVE	Swedish
THA	Thai
TUR	Turkish
UKE	UK English
ZHS	Simplified Chinese
ZHT	Traditional Chinese
COMMON	Nonlanguage attributes

---

## Available Release Numbers

The available release numbers are:

- Core 5.10.00.000
- Core 6.00.00.000
- Core 6.10.00.000
- Core 7.00.00.000
- Core 7.50.00.000
- Core 8.00.00.000
- Core 8.10.00.000
- Core 8.40.00.000
- Core.8.50.00.000



## Appendix C

# Using Default Style Classes

This chapter discusses how to use default style classes.

---

## Default Style Classes

This table lists many of the default style classes that come with PeopleSoft applications, a brief description of the style class, and the style sheet to which each class belongs:

<b><i>Style Class</i></b>	<b><i>Description</i></b>	<b><i>Style Sheet</i></b>
EDGE	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
EOPP_SCCHILDINDENT	Navigation collection child folder indentation.	PTSTYLEDEF
EOPP_SCPAGEBORDER	Navigation collection page border.	PTSTYLEDEF
EOPP_SCSECTIONCONTENTDARK	Navigation collection section content dark style.	PTSTYLEDEF
EOPP_SCSECTIONFOLDERDARK	Navigation collection section folder dark style.	PTSTYLEDEF
EOPP_SCSECTIONCONTENT	Navigation collection section content.	PTSTYLEDEF
EOPP_SCSECTIONFOLDER	Navigation collection section folder.	PTSTYLEDEF
EOPP_SCPAGETITLESECTION	Navigation collection page title.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
EOPP_SCBREADCRUMBSECTION	Navigation collection breadcrumbs.	PTSTYLEDEF
EOPP_SCADDITIONALTEXT	Navigation collection additional text.	PTSTYLEDEF
EOPP_SCPAGEDESCRSECTION	Navigation collection page description.	PTSTYLEDEF
EOPP_SCCHILDCONTENTLINK	Navigation collection child content link.	PTSTYLEDEF
EOPP_SCCHILDFOlderLINK	Navigation collection child folder link.	PTSTYLEDEF
EOPP_SCCHILDINDENT	Navigation collection child folder indentation.	PTSTYLEDEF
EOPP_SCMORELINK	Navigation collection more link.	PTSTYLEDEF
EOPP_SCPAGEBORDER	Navigation collection page border.	PTSTYLEDEF
EOPP_SCSECTIONCONTENT	Navigation collection section content.	PTSTYLEDEF
EOPP_SCSECTIONCONTENTDARK	Navigation collection section content dark style.	PTSTYLEDEF
EOPP_SCSECTIONFOLDERDARK	Navigation collection section folder dark style.	PTSTYLEDEF
EOPP_SCSECTIONCONTENTLINK	Navigation collection section content link.	PTSTYLEDEF
EOPP_SCSECTIONFOLDERLINK	Navigation collection section folder link.	PTSTYLEDEF
HEADERLINK	Used by Branding area links on a page.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
HEADERLINKACTIVE	Used by Branding area links on a page that are active.	PTSTYLEDEF
HILEFT1	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
HILEFT2	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
KNOCKOUT	Portal administration left navigation*.	PTSTYLEDEF
LEVEL1	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
LEVEL2	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
LEVEL3	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
MID	Links at the first level of the PeopleSoft Pure Internet Architecture navigation.	PTSTYLEDEF
MUTE	Descriptive text in the PeopleSoft Pure Internet Architecture navigation.	PTSTYLEDEF
PAADDITIONALINSTRUCTIONS	Additional instructions on the page (slightly smaller font).	PSALTERNATE
PABACKGROUNDINVISIBLE	Body background of hidden group boxes and grids.	PSALTERNATE
PABOLDBLUETEXT	Strong emphasis for text on a page.	PSALTERNATE

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PABOLDTEXT	Field value emphasis.	PSALTERNATE
PACALENDARDAYNUM	Text for numeric days on calendars.	PSALTERNATE
PACALENDARDAYNUMDISABLED	Text for numeric days on calendars (disabled state).	PSALTERNATE
PACFGRIDFOOT	Footer borders for ChartField grids.	PSALTERNATE
PACFGRIDLABEL	Header label background for ChartField grids.	PSALTERNATE
PAERRORTTEXT	Inline error message text font type, color, and size.	PSALTERNATE
PAEXAMPLE	Smaller font size to demonstrate an example.	PSALTERNATE
PAEXPANDCOLLAPSE	Expandable and collapsible area borders.	PSALTERNATE
PAFRAMEBACKGROUND	Same as the PAGROUPDIVIDER style class except that it allows text on the dividing bar for self-service pages.	PSALTERNATE
PAFRAMELEVEL1	Level 1 frames.	PSALTERNATE
PAFRAMELEVEL2	Level 2 frames.	PSALTERNATE
PAFRAMELEVEL3	Level 3 frames.	PSALTERNATE
PAGRIDHEADER	Grid column headers for self-service grids without navigation controls.	PSALTERNATE
PAGROUPBOX	Group boxes.	PSALTERNATE



<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PAGROUPBOXLABELINVISIBLE	Invisible group box label.	PSALTERNATE
PAGROUPBOXLABELLEVEL1	Group box labels at level 1.	PSALTERNATE
PAGROUPBOXLABELLEVEL2	Group box labels at level 2.	PSALTERNATE
PAGROUPBOXLABELLEVEL3	Group box labels at level 3.	PSALTERNATE
PAGROUPBOXLEVEL1	Group box body at level 1.	PSALTERNATE
PAGROUPBOXLEVEL2	Group box body at level 2.	PSALTERNATE
PAGROUPBOXLEVEL3	Group box body at level 3.	PSALTERNATE
PAGROUPDIVIDER	Group box label style on self-service pages.	PSALTERNATE
PAGROUPDIVIDERBODY	Group box body style on developer-defined advanced search pages.	PSALTERNATE
PAHEADERTOOLBARBODY	Frame background for the customer relationship management (CRM) toolbar.	PSALTERNATE
PAHEADERTOOLBARFRAME	Frame borders for the CRM toolbar.	PSALTERNATE
PAHORIZONTALRULELEVEL1	Horizontal rule at level 1.	PSALTERNATE
PAHORIZONTALRULELEVEL2	Horizontal rule at level 2.	PSALTERNATE
PAHORIZONTALRULELEVEL3	Horizontal rule at level 3.	PSALTERNATE
PAHYPERLINKDESCR	Text description for links on application homepages.	PSALTERNATE
PAHYPERLINKERRSTAT	Links conveying an error status (use with image).	PSALTERNATE

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PAHYPERLINKLVL1ODD	Link style for grid odd rows that use the PeopleCode style property.	PSALTERNATE
PAHYPERLINKSUCCESSSTAT	Links conveying a successful status (use with image).	PSALTERNATE
PAHYPERLINKWARNSTAT	Links conveying a warning status (use with image).	PSALTERNATE
PALEGENDBACKGROUND	Group box body color for legends on pages.	PSALTERNATE
PALEVEL0PRIMARY	Level 0, high-level key data on self-service pages.	PSALTERNATE
PALEVEL0SECONDARY	Additional fields to identify the object.	PSALTERNATE
PAPAGEINSTRUCTIONS	Text instructions on the top of the page.	PSALTERNATE
PAPAGETITLE	Page title.	PSALTERNATE
PASEARCHBOXBODY	Group boxes around search criteria.	PSALTERNATE
PASSINFORMATIONALTEXT	Text instructions on the top of the page.	PSALTERNATE_SWAN
PASSTRANSACTIONALTITLE	Transaction name above the current page title.	PSSTYLEDEF_SWAN
PASTEPBYSTEPSTITLE	Step-by-step transaction.	PSALTERNATE
PASUBTABLINKACTIVE	Active Sub tab style.	PSALTERNATE
PASUBTABLINKINACTIVE	Inactive Sub tab style.	PSALTERNATE
PATEXTBACKGROUND	Text background.	PSALTERNATE

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PATRANSACTIONTITLE	Transaction name above the current page title.	PSALTERNATE
PSACTIVETAB	Active tab labels and background colors.	PTSTYLEDEF
PSBOTTOMLINE	Tab bottom line style to specify height and width. Height can be set to 0 if tab bottom line is not required.	PTSTYLEDEF
PSCHARTAXISTITLE	Chart axis labels.	PTSTYLEDEF
PSCHARTDEFAULT	Default chart text.	PTSTYLEDEF
PSCHARTTITLE	Chart title text.	PTSTYLEDEF
PSCHECKBOX	Check box label.	PTSTYLEDEF
PSDISABLED	Disabled field value font type, size, and color.	PTSTYLEDEF
PSDROPDOWNLABEL	Drop-down list box label.	PTSTYLEDEF
PSDROPDOWNLIST	Drop-down list box text.	PTSTYLEDEF
PSEDITBOX	Edit box label.	PTSTYLEDEF
PSEDITBOX_DISABLED	Disabled edit box style (white background with a thin, gray border).	PTSTYLEDEF
PSEDITBOX_DISPONLY	Display-only edit boxes.	PTSTYLEDEF
PSEDITBOXLABEL	Edit box text.	PTSTYLEDEF
PSERROR	Error field background color and text.	PTSTYLEDEF
PSFRAME	Frame borders.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSGRIDCOLUMNHDR	Grid column header texts.	PTSTYLEDEF
PSGRIDCOUNTER	Color for the row counter text displayed in the grid header.	PTSTYLEDEF
PSGRIDEVENROWSORTED	Sorted even row style in a grid.	PTSTYLEDEF
PSGRIDODDROWSORTED	Sorted odd row style in a grid.	PTSTYLEDEF
PSGROUPBOX	Group box borders and body style.	PTSTYLEDEF
PSGROUPBOXLABEL	Group box labels.	PTSTYLEDEF
PSHEADERHYPERLINK	Link in a scroll area or grid header.	PTSTYLEDEF
PSHEADERHYPERLINKD	Disabled link in a scroll area or grid header.	PTSTYLEDEF
PSHEADERTOOLBARFRAME	Meant for PeopleTools usage for buttons.	
PSHORIZONTALRULE	Horizontal rule.	PTSTYLEDEF
PSHYPERLINK	Link.	PTSTYLEDEF
PSHYPERLINKACTIVE	Selected link.	PTSTYLEDEF
PSHYPERLINKDISABLED	Disabled link.	PTSTYLEDEF
PSHYPERLINKHOVER	Link when the mouse hovers over it.	PTSTYLEDEF
PSHYPERLINKNOUL	Link with no underline.	PTSTYLEDEF
PSHYPERLINKVISITED	Previously visited link.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSHYPERLINKYELLOWBKGRD	Link with a yellow background.	PTSTYLEDEF
PSIMAGE	Image border style, width, and colors.	PTSTYLEDEF
PSINACTIVETAB	Inactive tab labels and background colors.	PTSTYLEDEF
PSINACTIVETABHOVER	Inactive tab labels and background colors when the mouse hovers over them.	PTSTYLEDEF
PSLEVEL1GRID	Level 1 grid background colors, border settings, and fonts.	PTSTYLEDEF
PSLEVEL1GRIDACTIVETAB	Active tab labels and background colors on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDCOLUMNHDR	Grid column labels on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDCOLUMNHDRTXT	Level 1 grid column header text style	PTSTYLEDEF
PSLEVEL1GRIDEVENROW	Even rows on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDINACTIVETAB	Inactive tab labels and backgrounds on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDINACTIVETABHOVER	Inactive tab labels and backgrounds (when mouse hovers over them) on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDLABEL	Level 1 grid labels.	PTSTYLEDEF
PSLEVEL1GRIDNAVIGATIONBAR	Navigation bars on level 1 grids.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSLEVEL1GRIDODDROW	Odd rows on level 1 grids.	PTSTYLEDEF
PSLEVEL1GRIDROW	Level 1 grid background settings, border settings, and fonts.	PTSTYLEDEF
PSLEVEL1SCROLLAREA BODY	Level 1 scroll area body.	PTSTYLEDEF
PSLEVEL1SCROLLAREAFOOTER	Level 1 scroll area footer.	PTSTYLEDEF
PSLEVEL1SCROLLAREAHEADER	Level 1 scroll area header.	PTSTYLEDEF
PSLEVEL2GRID	Level 2 grid background colors, border settings, and fonts.	PTSTYLEDEF
PSLEVEL2GRIDACTIVETAB	Active tab labels and background colors on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDCOLUMNHDR	Grid column labels on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDCOLUMNHDRTXT	Level 2 grid column header text style.	PTSTYLEDEF
PSLEVEL2GRIDEVENROW	Even rows on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDINACTIVETAB	Inactive tab labels and backgrounds on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDINACTIVETABHOVER	Inactive tab labels and backgrounds (when mouse hovers over it) on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDLABEL	Labels on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDNAVIGATIONBAR	Navigation bars on level 2 grids.	PTSTYLEDEF
PSLEVEL2GRIDODDROW	Odd rows on level 2 grids.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSLEVEL2GRIDROW	Level 2 grid background settings, border settings, and fonts.	PTSTYLEDEF
PSLEVEL2SCROLLAREABODY	Level 2 scroll area body.	PTSTYLEDEF
PSLEVEL2SCROLLAREAFOOTER	Level 2 scroll area footer.	PTSTYLEDEF
PSLEVEL2SCROLLAREAHEADER	Level 2 scroll area header.	PTSTYLEDEF
PSLEVEL3GRID	Level 3 grid background settings, border settings, and fonts.	PTSTYLEDEF
PSLEVEL3GRIDACTIVETAB	Active tab labels and background colors on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDCOLUMNHDR	Grid column labels on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDCOLUMNHDRTXT	Level 3 grid column header text style.	PTSTYLEDEF
PSLEVEL3GRIDEVENROW	Even rows on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDINACTIVETAB	Inactive tab labels and backgrounds on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDINACTIVETABHOVER	Inactive tab labels and backgrounds (when the mouse hovers over them) on level 3 grids .	PTSTYLEDEF
PSLEVEL3GRIDLABEL	Labels on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDNAVIGATIONBAR	Navigation bars on level 3 grids.	PTSTYLEDEF
PSLEVEL3GRIDODDROW	Odd rows on level 3 grids.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSLEVEL3GRIDROW	Level 3 grid background settings, border settings, and fonts.	PTSTYLEDEF
PSLEVEL3SCROLLAREABODY	Level 3 scroll area body.	PTSTYLEDEF
PSLEVEL3SCROLLAREAFOOTER	Level 3 scroll area footer.	PTSTYLEDEF
PSLEVEL3SCROLLAREAHEADER	Level 3 scroll area header.	PTSTYLEDEF
PSLONGEDITBOX	Long edit box data.	PTSTYLEDEF
PSLONGEDITLABEL	Long edit box labels.	PTSTYLEDEF
PSMULTILANG	Multilingual field background.	PTSTYLEDEF
PSNAVPARENTLINK	Left navigation link color.	PTSTYLEDEF
PSPAGE	Page background color and margins.	PTSTYLEDEF
PSPAGECONTAINER	Background color for the page container.	PTSTYLEDEF
PSPAGECUST	Shaded background for page personalization image.	PTSTYLEDEF
PSPROCESSING	Processing message background color.	PTSTYLEDEF
PSPSMALLTEXT	Footer text font type, size, weight, and color.	PTSTYLEDEF
PSPTABTABLE	Style for Portal tab label.	PTSTYLEDEF
PSPUSHBUTTON	Push button labels and background.	PTSTYLEDEF



<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSPUSHBUTTON*	Style class for push buttons in non-SWAN mode. For example, .PSPUSHBUTTONRETURN, when clicked returns to the previous page.	PTSTYLEDEF
PSPUSHBUTTONDISABLED	Disabled push button labels and background color.	PTSTYLEDEF
PSRADIOBUTTON	Radio buttons.	PTSTYLEDEF
PSRIGHTLEFTSCROLL	Style specification for Show following Tabs and Show previous Tabs button.	
PSSRCHACTION	Font style for Search action.	
PSSRCHCHECKBOX	Search check box.	PTSTYLEDEF
PSSRCHDROPDOWNLABEL	Search drop-down labels.	PTSTYLEDEF
PSSRCHEDITBOXLABEL	Search edit box labels.	PTSTYLEDEF
PSSRCHINSTRUCTIONS	Search instructions.	PTSTYLEDEF
PSSRCHPAGE	Search page font type, background color, and margins.	PTSTYLEDEF
PSSRCHRESULTSEVENROW	Even rows on search page result grids.	PTSTYLEDEF
PSSRCHRESULTSFOOTER	Footer on search page result grids.	PTSTYLEDEF
PSSRCHRESULTSHDR	Header on search page result grids.	PTSTYLEDEF
PSSRCHRESULTSHDRTXT	Header text style for search results.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PSSRCHRESULTSHYPERLINK	Navigation links above the search results.	PTSTYLEDEF
PSSRCHRESULTSHYPERLINKD	Disabled navigation links above the search results.	PTSTYLEDEF
PSSRCHRESULTSODDROW	Odd rows on search page result grids.	PTSTYLEDEF
PSSRCHRESULTSTITLE	Instructional text on search results page.	PTSTYLEDEF
PSSRCHSUBTITLE	Search results section title.	PTSTYLEDEF
PSSRCHTIPS	Search tips.	PTSTYLEDEF
PSSRCHTITLE	First-level title on a search page.	PTSTYLEDEF
PSSTATICIMAGE	Static image border styles, width, and colors.	PTSTYLEDEF
PSTEXT	Static text controls.	PTSTYLEDEF
PSTLEFTMENU	No longer used.	PTSTYLEDEF
PSTREENODESELECTED	Selected tree node.	PTSTYLEDEF
PSTRELLINKSBODYNARROW	Related links body for narrow templates.	PAALTERNATE
PSTRELLINKSBODYWIDE	Related links body for wide templates.	PAALTERNATE
PSTRELLINKSHDRNARROW	Related links header label for narrow templates.	PAALTERNATE
PSTRELLINKSHDRWIDE	Related links header label for wide templates.	PAALTERNATE

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PTBREADCRUMB	Breadcrumbs.	PTSTYLEDEF
PTBREADCRUMBCUR	Current location on a breadcrumb trail.	PTSTYLEDEF
PTBREADCRUMBMARGIN	Breadcrumb margins.	PTSTYLEDEF
PTGRIDROWMOUSEOVER	Grid row style on mouse over.	PTSTYLEDEF
PTGRIDROWSELECT	Grid row style for a selected row.	PTSTYLEDEF
PT_MCFADDITIONALINSTRUCTIONS	Any additional instructions/error messages with small font.	PTSTYLEDEF
PT_MCFCHATME	Self chat message class.	PTSTYLEDEF
PT_MCFCHATSYSMSG	System chat message.	PTSTYLEDEF
PT_MCFCHATTHEM	Chat message from others.	PTSTYLEDEF
PT_MCFCHATUS	Chat message from another Agent.	PTSTYLEDEF
PT_MCFCHECKBOX	MCF check box.	PTSTYLEDEF
PT_MCFCHECKBOXLABEL	MCF check box label.	PTSTYLEDEF
PT_MCFDROPDOWNLIST	MCF drop-down list box.	PTSTYLEDEF
PT_MCFEDITBOX	MCF edit box.	PTSTYLEDEF
PT_MCFEVTNAME	REN event name in tracer log.	PTSTYLEDEF
PT_MCFEVTVALUE	REN event value in tracer log.	PTSTYLEDEF
PT_MCFGRAYTEXT	Non-editable gray text.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PT_MCFGROUPBOX	MCF group box style.	PTSTYLEDEF
PT_MCFHEADER	MCF header style.	PTSTYLEDEF
PT_MCFLABEL	MCF label style.	PTSTYLEDEF
PT_MCFPAGETITLE	MCF page title style.	PTSTYLEDEF
PT_MCFPUSHBUTTON	MCF push button style.	PTSTYLEDEF
PT_MCFROW	Row style in an MCF table.	PTSTYLEDEF
PT_MCFSPVSRTABLE	Table style for MCF monitoring supervisor.	PTSTYLEDEF
PT_MCFTABLE	Table style.	PTSTYLEDEF
PT_MCFTABLEHEADER	Transparent table header style.	PTSTYLEDEF
PT_MCFTABLEHEADER1	Table header style.	PTSTYLEDEF
PT_MCFTRACEREVENT	REN event in tracer window style	PTSTYLEDEF
PT_MCFTRACERREQ	REN request in tracer window style.	PTSTYLEDEF
PTCXMADMINBAR	Context manager "Administer Pagelets" bar style.	PTSTYLEDEF
PTCXMADMINLINK	Context Manager "Administer Pagelets" link style.	PTSTYLEDEF
PTCXMPAGELETBODYBAK	Context Manager template pagelet body background.	PTSTYLEDEF
PTCXMPAGELETHEADERBAK	Context Manager template pagelet header background	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PTCXMPGLTHDRLINK	Context Manager template pagelet header link.	PTSTYLEDEF
PTCXMPGLTHDRLINKHOVER	Context Manager template pagelet header link hover.	PTSTYLEDEF
PTGRIDROWMOUSEOVER	Grid row style on mouse over.	PTSTYLEDEF
PTGRIDROWSELECT	Grid row style for a selected row.	PTSTYLEDEF
PTMENULINK	Link on level 2 of the PeopleSoft Pure Internet Architecture navigation.	PTSTYLEDEF
PTNAVBACKGROUND	Background style in navigation menu.	PTSTYLEDEF
PTNAVLINK	Content reference link style.	PTSTYLEDEF
PTNAVLINKHOVER	Content reference link style when the mouse hovers over it.	PTSTYLEDEF
PTNAVSEARCHDIVIDER	Search divider element style (in navigation).	PTSTYLEDEF
PTNAVSEARCHLABEL	Search label style (in navigation).	PTSTYLEDEF
PTNAVSELCHILDBK	Selected content reference background color (in navigation).	PTSTYLEDEF
PTNAVSELCHILDEDGEDK	Background dark color on the top edge of the selected content reference (in navigation).	PTSTYLEDEF
PTNAVSELCHILDEDGELT	Background color on the top edge of the selected content reference (in navigation).	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PTNAVSELCHILDLINK	Selected content reference link font type and color (in navigation).	PTSTYLEDEF
PTNAVSELFOLDERBK	Selected folder background color (in navigation).	PTSTYLEDEF
PTNAVSELPARENTLINK	Selected parent link color (in navigation).	PTSTYLEDEF
PTNAVSELPARENTBK	Selected parent folder background color (in navigation).	PTSTYLEDEF
PTNAVSELPARENTEDGE	Background color on the edge of the selected parent folder (in navigation).	PTSTYLEDEF
PTPAGELET	Homepage pagelet border thickness and color.	PTSTYLEDEF
PTPAGELETBODY	Homepage pagelet font type and size, border thickness, border color, and padding.	PTSTYLEDEF
PTPAGELETHEADER	Homepage pagelet header font type, size, and color; background color, border thickness; border color; and padding.	PTSTYLEDEF
PTPOPUP_HEADER	Header style for pop up page element.	PTSTYLEDEF
PTPOPUP_TITLE	Title style that appears in the header of the pop-up page element.	PTSTYLEDEF
PTQUERYSTYLESUB	Query style definition.	PTSTYLEDEF
PTSELECTBODY	Portal layout page selection box body.	PTSTYLEDEF

<b>Style Class</b>	<b>Description</b>	<b>Style Sheet</b>
PTSELECTHEADER	Portal layout page selection box header.	PTSTYLEDEF
PTSRCHRESULTSHDRTXT	Text style in Search results header.	PTSTYLEDEF
PTSPELLCHECKERROR	Invalid word for the spell check (highlighted in black with white text).	PTSTYLEDEF
PTTRANSPARENT	Displays text labels as transparent so that they are not visible in standard layout mode but are embedded in the HTML so that they can be read by a screen reader.	PTSTYLEDEF
RESET	Default PeopleSoft Pure Internet Architecture navigation*.	PTSTYLEDEF
SMALL	Link on the portal homepage personalization page and portal structure breadcrumbs.	PTSTYLEDEF
STRONG	Emphasized label text size and weight.	PSSTYLEDEF
STRONGWRAP	Emphasized text size and weight that will break to the next line in long sentences.	PTSTLYEDEF_SWAN

\* Default PeopleSoft Pure Internet Architecture styles are no longer in use.

---

**Note.** Style classes with the suffix *WBO* have border properties. For example, the style `PSGROUPBOXWBO` enables you to define border properties such as color, width, and style under the Border tab of the style property.

Style classes with the suffix *NBO* have no border properties. For example, the style `PSGROUPBOXNBO` enables you to define a border property having *None* as its style.

---





## Appendix D

# Modifying Classic Style Sheet Tabs

This chapter provides an overview of classic tab definitions and discusses how to modify classic style sheet tabs.

---

## Understanding Classic Tab Definitions

You can change the background colors of folder tabs in a component and tabs in a grid. To change the background colors of tabs, you should have a basic understanding of how tab definitions are built and stored in the database.

PeopleSoft applications store tab definitions as images. Each tab, or image definition, has three parts that are stored as rows in the database:

- Row 1 contains code representing the top border.
- Row 2 contains the tab itself.
- Row 3 contains the bottom border.

Because tabs look different, depending on where they are located relative to other tabs and whether they are active, you must define them in sets for a combination of active and inactive tab colors.



Active General tab with inactive tabs

The tab image names reflect the role of the image in the tab row and the colors that it uses. Colors are encoded using the 6-character, hexadecimal RGB value of the color.

When the system displays tabs on a page, it verifies the existence of all of the images that are required for the pair of colors that is specified by the active and inactive tab styles. If the system finds these images, the tabs are drawn using these colors. If the system cannot find the images, it uses white (FFFFFF) for the active tab and blue (6666FF) for the inactive tab. If the images for these colors are not in the database, then the tabs do not appear at all.

### ***Classic Tab Image Naming Scheme***

This topic details how classic tab images are named. However, to change tab colors, change only the RGB values.

All tab images are named in this format:

**PT\_TABNABBRRGGBB**

- **PT\_TAB**: System-defined image definition name prefix.
- **N**: HTML table row.

Values are 1, 2, or 3. Create images only for 2 or 3.

- **A**: Location of the tab parts in a tab.

Values are:

- **L**: Left end.
- **B**: Between.
- **R**: Right end.
- **M**: Middle of a tab.
- **BB**: Relative position to other tabs.

Values are:

- **AI**: Piece between active and inactive.
- **IA**: Piece between inactive and active.
- **II**: Piece between two inactive tabs.
- **AX**: Piece of an active tab.
- **IX**: Piece of an inactive tab.
- **RRGGBB** is either omitted (when the tab image does not use any color), or is one or two 6-character hexadecimal codes for inactive and active tab colors.

---

## Modifying Classic Style Sheet Tabs

This section discusses how to create tab images.

### Creating Tab Images

Before you can assign a new background color to the active or inactive tabs, you must create the necessary tab images with that color.

To produce tabs in a different color combination:

1. Open the image definitions for the tabs that you want to change.

Because you want to change only the color of tabs, open only those image definitions that already contain an RGB color code.

If you are changing both the active and inactive tab color, access these image definitions:

- PT\_TAB2LAXFFFFFF
- PT\_TAB2RAXFFFFFF
- PT\_TAB2LIXB8B090
- PT\_TAB2RIXB8B090
- PT\_TAB2BAIB8B090FFFFFF
- PT\_TAB2BIAB8B090FFFFFF
- PT\_TAB2BIIB8B090
- PT\_TAB3LAXFFFFFF
- PT\_TAB3RAXFFFFFF
- PT\_TAB3MAXFFFFFF
- PT\_TAB3LIXB8B090
- PT\_TAB3RIXB8B090
- PT\_TAB3MIXB8B090
- PT\_TAB3BAIB8B090FFFFFF
- PT\_TAB3BIAB8B090FFFFFF
- PT\_TAB3BIIB8B090

If you are changing only the inactive color, which is beige, you only need the files that contain the beige color code, B8B090. Similarly, if you are changing only the active tab color, which is white, you only need the files that include FFFFFFFF. However, for the changes to apply, you must change all those files with that color code. If one file is missed or a file name has a typographical error, the default style still applies.

2. To export the file to a directory, select File, Export Image.
3. In the Save Image dialog box, select the directory in which you want to place the image.
4. Use any image editor to alter the image colors.

---

**Note.** Do not change the size or shape of the images and do not replace any of the transparent pixels with solid pixels.

---

5. In PeopleSoft Application Designer, add the new image definitions by selecting File, New, Image.

Save the new image files under the appropriate name. The new images must have new names reflecting the HTML colors that they represent. For example, replace FFFFFFFF with the new active tab color, and replace B8B090 with the new inactive tab color. Be careful when naming these images. If the images with the correct colors for the tab style are not in the database, the system automatically applies the default images.

6. Open the appropriate style sheet.

If you are changing the tab folder colors, use PSSTYLEDEF. If you are changing grid tabs, use PSSTYLEDEF or the style sheet that is associated with the page definition in which the grid is located.

---

**Note.** You can set the grid tab styles for each grid individually using the grid properties. However, for page tabs, you can set the styles using only the default styles in the application style sheet.

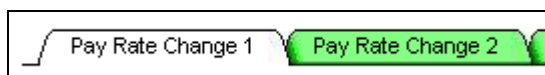
---

You must access PSACTIVETAB, PSINACTIVE TAB, and any other tab classes that your application uses (for example, PSLEVEL1GRIDACTIVETAB).

7. For folder tabs, open the PSACTIVETAB class by double-clicking the class name.
8. On the Background tab, select the color of the background of the active tab (matching the color of the tab image that you created).  
  
Use the drop-down list box to find the color or browse through the color choices.
9. Do the same for the PSINACTIVETAB class (or any others that you need) for folder tabs by double-clicking the class name and selecting the Background tab.
10. For grid tabs, access the style class that is associated with the grid tab.
11. Change the colors as indicated in the previous steps.
12. Save the style sheet.

## Examples

This is an example of folder tabs with a green background. In addition, the color of the text on the tabs was changed to black:



Example of green tabs

In the example, the active tab color stayed the same (FFFFFFF). Only the inactive tab color changed. Therefore, only these files were created and added to the list of images:

- PT\_TAB2LIX80FF80
- PT\_TAB2RIX80FF80
- PT\_TAB2BAI80FF80FFFFFFF
- PT\_TAB2BIA80FF80FFFFFFF

- PT\_TAB2BII80FF80
- PT\_TAB3LIX80FF80
- PT\_TAB3RIX80FF80
- PT\_TAB3MIX80FF80
- PT\_TAB3BAI80FF80FFFFFF
- PT\_TAB3BIA80FF80FFFFFF
- PT\_TAB3BII80FF80



## Appendix E

# Using Images in PeopleSoft Applications

This chapter provides an overview of the five image-related features of PeopleSoft Application Designer and discusses how to:

- Use PeopleTools image definitions.
- Generate a list of images in the image catalog.
- View all images in the image catalog.

---

## Understanding Image-Related Elements

You find images throughout PeopleSoft applications. Images serve many purposes such as identifying an employee or indicating to a user their current step in a series of steps in a process. PeopleSoft Application Designer offers several image-related elements, each of which you use depending on the purpose of the image. The following table describes the usage of each image-related element and provides an example of it:

<i><b>Image-Related Feature</b></i>	<i><b>Usage</b></i>	<i><b>Example</b></i>
Image field	Use to create a field in a database table that stores images in a user-defined format, such as portable network graphic (PNG) or bitmap (BMP). Like all fields, you place image fields in record definitions.	The EMPLOYEE_PHOTO field is an image field in the EMPLOYEE_PHOTO record definition.
ImageReference field	Use to create a field in a derive/work table that temporarily holds an image that can change programmatically at runtime using PeopleCode. Like all fields, you place image fields in record definitions.	The LOGO field is an image reference field that appears in the TE_HTL_LIST derived work record definition.
Image definition	Use to store an image file as a PeopleTools-managed item in the image catalog.	For example, the spinning image that appears when the system processes user information is the PT_PROCESSING image definition

<i>Image-Related Feature</i>	<i>Usage</i>	<i>Example</i>
Image control	Use to display image data that is stored in image fields in the database.	For example, an image control appears on the EMPLOYEE_PHOTO page; this control references the EMPLOYEE_PHOTO record field.
Static image control	Use to display a static image that is stored in the image catalog.	For example, a static image control appears on the COMPANY_CAR page; this control references the CAR_STOCK image in the image catalog.

### See Also

[Chapter 4, "Creating Field Definitions," Specifying Image Field Attributes, page 73](#)

[Chapter 4, "Creating Field Definitions," Specifying ImageReference Field Attributes, page 73](#)

[Chapter 9, "Using Page Controls," Setting Image Properties, page 220](#)

[Chapter 9, "Using Page Controls," Setting Static Image Properties, page 219](#)

## Using PeopleTools Image Definitions

All PeopleSoft applications include a common set of PeopleTools images that in the image catalog. If you use PeopleTools images when developing your own applications, you should use each image in a way that is consistent with its use in PeopleSoft applications.

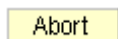
---

**Important!** Do not change PeopleTools image definitions. If you want to change a PeopleTools image definition, copy it and then modify the copy.

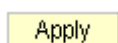
---

The following list contains many common PeopleTools images and their standard usage; the list is not exhaustive. The images are categorized by function and listed alphabetically by image name.

### General Images



PT\_ABORT: Abort a transaction. Used in some PeopleCode message boxes.



PT\_APPLY: Apply changes that were made to the database. Usually found when transferring to another page through PeopleCode.

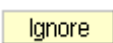


PT\_CALENDAR: Open the calendar prompt.

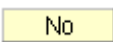




PT\_COPYURL: Used in the pagebar. Copy contents of address bar to the clipboard.



PT\_IGNORE: Used in some PeopleCode message boxes.



PT\_NO: Used in some PeopleCode message boxes.



PT\_OK: Accept the input on a secondary page and return to the main page.



PT\_POPUP: Open a pop-up menu from which the user can select a page to which to transfer.



PT\_POPUP\_SKIP: Use if a pop-up menu has only one menu item to select from; bring the user directly to a page.



PT\_PRINT: Open the print dialog box for the designated item.



PT\_PROCESSING: Appears as animated image while accessing a server or database.



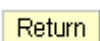
PT\_PROMPT\_LOOKUP: Lookup button for a field prompt table.



PT\_SPELLCHECK: Opens the spell check page for the current field.



PT\_REPORT\_DIST\_ICN: Usually found on the Process Scheduler Request page, this button takes you to the Distribution Detail page. Use to select roles and users who can access or receive an email of report output.



PT\_SEC\_RETURN: Return from a secondary page to the main (calling) page.



PT\_WF\_ACTIVITY: Used for a subprocess of the business task for workflow, consisting of one or more steps.



PT\_WF\_BUSPROC: Depict a complete business task for workflow.



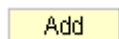
PT\_WF\_STEP: Show a discrete step in the business process that corresponds to a single transaction that is performed on an application page or through an external program.

### ***Scroll or Grid Action Images***

Use these images when creating scroll areas, scroll bars, and grids.



PT\_ADD: Add a row button.



PT\_ADD\_LARGE: Add a large row button. Alternative to PT\_ADD if the page has enough room.



PT\_DELETE: Delete a row button.



PT\_DELETE\_LARGE: Delete a large row button. Alternative to PT\_DELETE if the page has enough room.



PT\_GRID\_NO\_TABS: Expand grid button. Expand grid columns horizontally so that grid tabs no longer show.



PT\_GRID\_SHOW\_TABS: Tabbed grid button. Collapse grid columns horizontally so that grid tabs show.



PT\_NEXTROW: View the next row button.



PT\_PREVIOUSROW: View the previous row button.

### ***Tab Images***

Use these images to change the look of both folder and grid tabs.



PT\_TAB\_LSCROLL: Scroll left to the previous tab in a page or grid.



PT\_TAB\_RSCROLL: Scroll right to the next tab in a page or grid.



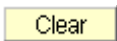
PT\_TABxxx through PT\_TABxxxxxxxx: Selection of more than 80 images you can use to create folder tabs.

### ***Lookup and Search Page Images***

Use these images on lookup and search pages.



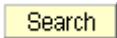
PT\_CANCEL: Cancel a lookup page.



PT\_CLEAR: Clear search criteria from the page (appears only on an Advanced Search/Lookup page).



PT\_LOOKUP: Begin looking up data.



PT\_SEARCH: Start a search on the search page.

### ***Query Images***



PT\_QUERY\_ADD\_CRITERIA: Add criteria to a query.



PT\_QUERY\_KEY: Denotes a query key field.



PT\_QUERY\_SORT: Sort query fields alphabetically.

## Tree Images

Use these tree images with PeopleSoft Tree Manager.

Tree node and leaf images include:



PSTREEMGR\_COL\_BRANCH: Identifies the root node for a branch.



PSTREEMGR\_COL\_NODE: Standard node image.



PSTREEMGR\_COL\_SKNODE: Tree node that has skipped a level.



PSTREEMGR\_END\_NODE: Tree node that has no children; that is, a terminal node.



PSTREEMGR\_EXP\_BRANCH: Root node on a branched tree.



PSTREEMGR\_EXP\_NODE: Expanded node.



PSTREEMGR\_EXP\_SKNODE: Expanded node that has skipped a level.



PSTREEMGR\_LEAF: Lowest-level detail value of a tree.

Node and leaf command button images include:



PTTREE\_ADDCHILD: Add a new child node.



PTTREE\_ADDLEAF: Add a new detail value, such as a leaf.



PTTREE\_ADDSIB: Add a new node as a sibling to the selected node.



PTTREE\_CUT: Cut a node or leaf and place it on the clipboard.



PTTREE\_DELETELEAF: Delete a leaf.



PTTREE\_DELETENODE: Delete a node.



PTTREE\_DISP\_AS\_ROOT: Redisplay the tree, starting with the currently selected node being displayed as the root node.



PTTREE\_EDITDATA: Display the underlying user data page.



PTTREE\_MAKEBRANCH: Create a new tree branch, starting with the selected node.



PTTREE\_PASTECHILD: Paste the node on the clipboard as a child of the currently selected node.



PTTREE\_PASTESIB: Paste the node on the clipboard as a sibling of the currently selected node.



PTTREE\_UNBRANCH: Unbranch a branch.



PTTREE\_UPDATELEAF: Update the values and properties for a leaf.



PTTREE\_UPDATENODE: Update the value or properties for a node.

### Portal Images

Use these images to perform a specific action. Additional portal images, such as PT\_PORTAL\_HEADER\_BG, are purely aesthetic. In general, all images that are reserved for the portal contain the word *PORTAL* in the image name, such as PT\_PORTAL\_SEPARATOR.



PT\_PORTAL\_DOWN\_ARROW\_Y: Move a pagelet down when personalizing a portal layout.



PT\_PORTAL\_IC\_CLOSE: Remove a pagelet from a homepage.

### Toolbar Button Images

Toolbar buttons are HTML buttons, not images. HTML defines the buttons and the text on the buttons. To modify toolbar buttons, use a style class.

See [Chapter 15, "Creating Style Sheet Definitions," Setting Style Class Attributes, page 454.](#)

---

**Note.** Icons that appear on toolbar buttons are images.

---

## Generating a List of Images in the Image Catalog

To generate a list of images in the image catalog, you can:

- In PeopleSoft Application Designer, select File, Open, Image and click the Open button.
- In the query tool for your database platform, enter and run this SQL statement to list all image definitions:

```
SELECT DISTINCT CONTNAME FROM PSCONTDEFN WHERE CONTTYPE = 1.
```

- In the query tool for your database platform, enter and run this SQL statement to the language-dependent versions of images, :

```
SELECT CONTNAME, LANGUAGE_CD, CONTTYPE CONTFMT DESCR FROM  
PSCONTDEFNLANG WHERE CONTTYPE = 1
```

---

**Note.** Before signing into PeopleSoft Application Designer in a different language, you must enable the language support. Use PeopleTools Utilities to enable support.

---

- Create a query definition in PSQuery that uses the PSCONTDEFN record, the CONTNAME field from that record, and select for images where the CONTTYPE = 1 only.

---

## Viewing All Images in the Image Catalog

You might want to see the images that exist in the image catalog. PeopleSoft Application Designer enables you to see some of the image definitions. However, you have to open each image definition. You can view all images online by creating a custom component. The elements of the component should include:

1. A record definition that contains one long character field.
2. A page definition that contains an HTML area you associate to the record field.
3. A SQL definition that retrieves the names of all image definitions in the image catalog.
4. A PeopleCode program that generates text in HTML format.

The program should generate the HTML by looping through and enclosing each image definition name in the appropriate HTML and metaHTML tags. The program can also add the image name and a line break so that the page is crisper. The program must concatenate the lines together to form the HTML as a string. Each line of HTML should look similar to this line:

```
| "<img src='%Image(" |&ImageDefinitionName| ")' />" |&ImageDefinitionName| "<BR />"
```

Finally, the program must assign the string output to the HTML record field that you associated to the HTML area.

After you create the component, you must register it and grant the appropriate permissions. Then you can access the page and view the images.



## Appendix F

# Performing Bulk Operations

This chapter provides an overview of bulk operations and discusses how to perform bulk:

- Insertions.
- Modifications.
- Deletions.

---

## Understanding Bulk Operations

PeopleTools enables administrators to perform bulk operations on fields that might be in many different records or pages.

---

**Note.** For security reasons, only those users logged in to PeopleSoft Application Designer using the PeopleSoft Administrator role have access to the Bulk Operations menu items. For all other users, these menu options are disabled. To gain access, navigate to PeopleTools, Security, User Profiles, User Profiles, Roles tab.

---

Bulk operations are useful for financial ChartFields. *ChartField* is an accounting term that describes a small group of character fields, typically 5 to 20, that are present in many records and pages. ChartFields can be between 1 and 30 characters long, with an average length of 10. They often identify organization-wide attributes, such as DEPTID (department identification).

---

**Important!** Many PeopleSoft financial applications have a standardized way to modify ChartFields by using standard pages and their corresponding PeopleCode application programming interfaces (APIs). The financials applications do more than just invoke the bulk operations to modify ChartFields. For this reason, users of these applications should follow the applications guidelines rather than using these bulk operations menu items in PeopleSoft Application Designer. Consult your application to ensure that you are using the appropriate mechanism to update your ChartFields before using the PeopleSoft Application Designer Bulk Operations tool.

---

---

## Performing Bulk Insertions

To perform bulk insertions using PeopleSoft Application Designer, select Tools, Bulk Operations, Insert.

This section lists common elements and discusses how to insert:

- Fields into records.

- Record fields onto pages.
- Fields into records and onto pages.

---

**Note.** If you try to insert a field that is already present in a record or on a page, no processing occurs.

---

## Common Elements Used in This Section

<b>Model Field</b>	Specify a field with record attributes that you want to copy to all inserted fields.
<b>Clone PeopleCode of model field</b>	Select if you want PeopleCode in the record fields to be copied and references modified to the inserted field.

## Inserting Fields into Records

- To perform bulk insertions using PeopleSoft Application Designer, select Tools, Bulk Operations, Insert.
- You can insert a source field into selected records only if the model field exists in those records. The source field is assigned the same record field properties as the model field in each record and is inserted immediately following the model field.
- If the model field has a prompt table, then the system creates a prompt table for the source field by using the name of the source field with *TBL* appended to it.
- If the record is either a SQL view or a dynamic view type, then the system modifies the associated SQL by expanding the SELECT clause to include the new field.
- If the record is a subrecord, then the system updates the parent records of the SQL view or dynamic view type that contain this subrecord. If the SQL contains the model field in the WHERE clause or if the SQL is complex, then the system inserts the associated record into a project called BLK\_<SOURCEFIELDNAME> and displays a status message. Examine the contents of that project after the operation completes to verify successful completion.
- If the model field has PeopleCode associated with it in the record or in a component, then the system copies this PeopleCode to the new field with all of the references to the model field changed to refer to the new field.

---

**Note.** Because performing this operation changes records that are used to build application tables, you must rebuild (alter) the database before these changes can be used.

---

## Inserting Record Fields onto Pages

- To perform bulk insertions by using PeopleSoft Application Designer, select Tools, Bulk Operations, Insert.
- This operation inserts the specified record field onto each page that is listed if the model field exists on the page.
- If the model field is in a grid, the system inserts the new field into the grid next to the model field and assigns it the same page field properties.



If the model field is not in a grid, the system inserts the new field to the right of the model field (in the first open space) and assigns it the same page field properties. If the system detects a questionable field position, it inserts the page into a project called BLK\_<SOURCEFIELDNAME> and displays a status message. Examine that project after the operation completes to verify successful completion.

The page field name property is not cloned if it exists on the model field. Instead, the system uses the name of the new field because the page field name should be a unique identifier for page elements.

---

**Note.** You do not need to rebuild the database after performing this operation. The changes take effect when you open the page.

---

**Record Name** Enter the record name containing the field to insert onto the pages.

**Get Pages from Project** Specify the project name containing the pages to modify.

**Get Records from Project** Specify the project containing the records to modify.

## Inserting Fields into Records and onto Pages

To perform bulk insertions by using PeopleSoft Application Designer, select Tools, Bulk Operations, Insert.

This operation is a combination of inserting fields into records and inserting record fields onto pages. The system examines each record to see if a model field exists. If so, it inserts the source field (including all field properties, PeopleCode, and SQL additions to itself and parent records) after the model field. Then, for each record, the system checks all pages to see if the new record field should be inserted.

If the system detects a questionable field position, it inserts the page into a project called BLK\_<SOURCEFIELDNAME> and displays a status message. Examine that project after the operation completes.

**Get Records and Pages from Project** Specify the project containing the records and pages to modify.

---

## Performing Bulk Modifications

To perform bulk modifications by using PeopleSoft Application Designer, select Tools, Bulk Operations, Modify.

This section discusses how to:

- Modify record fields.
- Modify page fields.
- Reorder or resize grid fields.
- Change page field attributes.

## Modifying Record Fields

To perform bulk modifications by using PeopleSoft Application Designer, select Tools, Bulk Operations, Modify.

Use this operation to modify record fields by applying record field properties to a field in a set of records. All of the properties that are associated with the page are set for the record field in the listed records. Therefore, after this operation completes, the source field in these records contains identical record field properties.

---

**Note.** In most cases, you must rebuild the database for these attributes to take effect.

---

<b>Field will be modified on records from Project</b>	Specify the project name containing the record fields to modify.
<b>Records</b>	Specify the records containing the fields with attributes that you want to modify.
<b>Change Attributes</b>	Click to open the menu to select specific attributes. Enter the appropriate attributes for the modified fields.

## Modifying Page Fields

To perform bulk modifications by using PeopleSoft Application Designer, select Tools, Bulk Operations, Modify.

Use this operation to modify page fields in a variety of ways. You can specify a source field list from a list of fields, records, pages, or projects. You can only modify page field attributes on a single field at a time; therefore, if you do not specify a single field, the Change Attributes button is unavailable.

<b>Field,From Record, From Page, and From Project</b>	Specify the source that is used to populate the Fields to be modified list.
<b>Get Pages from Project</b>	Specify the project that is used to populate the Pages to be modified list.
<b>Reorder/Resize Grid Fields</b>	Click to access the Modify Order of Page Grid Fields dialog box and specify the order and width of the Fields to be modified fields on all listed pages.
<b>Change Attributes</b>	Click to access the Modify Page Field Attribute dialog box and specify the attributes of the field on all listed pages.

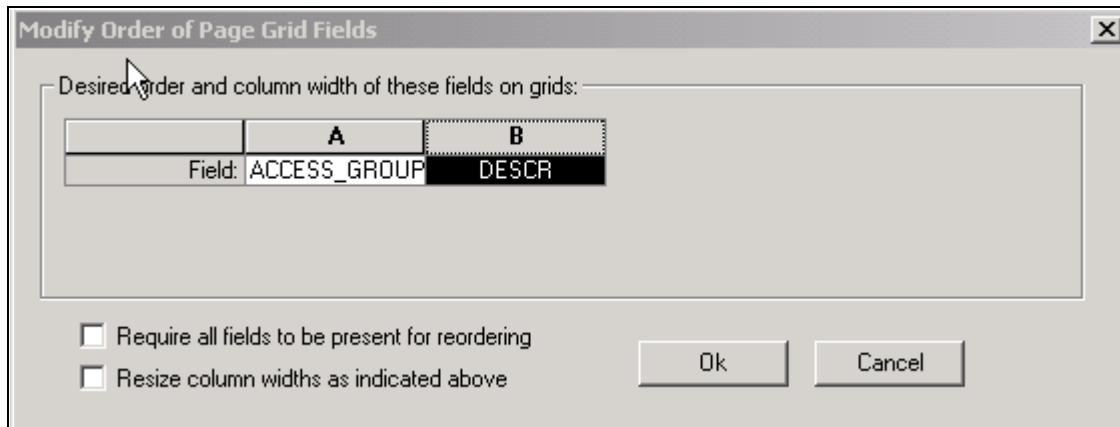
## Reordering or Resizing Grid Fields

To perform bulk modifications by using PeopleSoft Application Designer, select Tools, Bulk Operations, Modify.

You can rearrange columns as needed, and the order is applied on target pages. The reordering algorithm groups the fields together at the first instance of any of the fields in a target page grid and forces the remaining fields into the specified order.

Pages that have these fields outside of grids are not affected by these changes. The resize operation applies only to grid column widths.

This example shows the Modify Order of Page Grid Fields dialog box:



Modify Order of Page Grid Fields dialog box

**Require all fields to be present for reordering**

Select if you do not want to modify pages that do not have all of the fields in the grid.

**Resize column widths as indicated above**

Select to resize the columns to the column widths that are defined.

---

**Note.** These changes take effect immediately; no database build is required.

---

## Changing Page Field Attributes

To perform bulk modifications by using PeopleSoft Application Designer, select Tools, Bulk Operations, Modify.

Like record field attributes, you can modify page field attributes for the selected field (on the selected set of pages) by using the Modify Page Field Attribute dialog box. The system applies all of the attributes on the screen to the page field on all pages.

---

## Performing Bulk Deletions

To perform bulk deletions by using PeopleSoft Application Designer, select Tools, Bulk Operations, Delete.

Use the delete operation to:

- Remove the field from a list of pages, regardless of where the field exists on the page or whether it is in a grid.
- Remove the field from the list of records.

The delete operation removes associated PeopleCode and modifies associated SQL either in the record or, if the record is a subrecord, in parent records.

If the field is in the SELECT clause of the SQL, the removal is straightforward. However, if the field is also used in a WHERE clause or if the field is the only item in the SELECT clause, then the record is not modified. Instead, it is inserted into a project called BLK\_<SOURCEFIELDNAME> as noted in messages in the status screen.

---

**Note.** Users should examine a project with a targeted field in SQL statements individually to delete the field as required from the associated SQL.

---

Deleting fields from records and pages does *not* remove the field definition itself and does not remove the field from other applications, such as Crystal Reports.

---

**Note.** Because performing this operation changes records, you must subsequently rebuild the database because each application data table is based on the record definition.

---

## Appendix G

# Running PeopleTools Cross-Reference Reports

This chapter provides an overview of cross-reference reports and discusses how to run Crystal reports.

---

## Understanding Cross-Reference Reports

Using PeopleTools, you can create new applications by defining menus and pages that you use to enter data and database tables. PeopleTools includes a variety of cross-reference reports (similar to blueprints) that guide you through the definitions (such as menus, pages, and records) that are delivered with the system and those that you create or adapt with PeopleTools.

PeopleTools cross-reference reports are predefined Crystal reports, not unlike the standard reports that are delivered with the PeopleSoft application. Like other standard reports, you can identify cross-reference reports by the three-character prefix XRF. The reports provide several views of the application, ranging from high-level lists of the windows, menus, and pages to the detailed database attributes of fields.

You should already be familiar with the types of definitions that are delivered with the PeopleSoft application, as well as with PeopleTools and relational databases.

This table describes the cross-reference reports:

<i><b>Report</b></i>	<i><b>ID</b></i>	<i><b>Description</b></i>
Fields and Pages	XRFFLPN	Lists all fields in alphabetical order. It includes the names of all record and page definitions in which the fields occur and lists the field long names.
Fields and Records	XRFRCL	Lists all fields in alphabetical order by the associated record definition name. It details the field long name, type, length, and formatting.
Records and Fields	XRFFLRC	Lists all fields in alphabetical order. It includes the field long name, type, length, and formatting, as well as the names of all record definitions that contain the field.

<b>Report</b>	<b>ID</b>	<b>Description</b>
Applications and Fields	XRFAPFL	Lists all menus, such as general tables, in alphabetical order, including the fields in each menu. For each field, it lists the field name, type, length, and format, as well as all record and page definitions that contain the field (in the window).
Fields Referenced by PeopleCode Programs	XRFFLPC	Lists all PeopleCode programs in alphabetical order by the associated record definition and field. It includes the types of fields that are referenced in the PeopleCode program.
Field Listing	XRFIELDS	Lists all fields in alphabetical order. It includes field type, length, format, long name, and short name.
Menu Listing	XRFMENU	Lists all menus in alphabetical order, including all page definitions in each menu. It includes the associated search record definition name and detail page definition name.
Page Listing	XRFPAGE	Lists all page definitions in alphabetical order.
PeopleCode Programs and Field References	XRFPCFL	Lists record definitions that contain fields with PeopleCode program attributes. It includes the field name, as well as the associated record definitions and fields that are referenced in the PeopleCode program.
Pages with PeopleCode	XRFPNPC	Lists all pages that contain fields with PeopleCode attributes. For each page, it includes the name of the record definitions that contain the field, as well as the field name and type.

<b>Report</b>	<b>ID</b>	<b>Description</b>
Records and Pages	XRFRCPN	Lists all record definitions in alphabetical order. It includes the menu and page definitions that are associated with each record definition.
Window Listing	XRFWIN	Lists all application menu windows in alphabetical order.

---

**Note.** In addition to standard cross-reference reports, if you are familiar with the PeopleSoft database you can generate additional ad hoc reports to extract the exact combination of information that you need.

---

---

## Running Crystal Reports

This section discusses how to generate Field Cross Reference (XRFIELDS) reports.

You run cross-reference reports on the Process Scheduler Request page.

Access the Process Scheduler Request page.

1. Select PeopleTools, Process Scheduler , System Process Requests.

The Sample Processes search page appears.

2. Click Add a New Value.
3. Enter *CrystalClient* for the run control ID.
4. Click Add.

5. Click Run.

The Process Scheduler Request page appears.

Process Scheduler Request

User ID: QEDMO

Run Control ID: CrystalClient

Server Name: PSNT

Run Date: 08/11/2011

Recurrence:

Run Time: 5:32:23PM

Reset to Current Date/Time

Time Zone:

Process List

Select	Description	Process Name	Process Type	*Type	*Format	Distribution	File Dependency
<input type="checkbox"/>	<a href="#">COBOL Multi-process Job</a>	3CBL	PSJob	(None)	(None)	<a href="#">Distribution</a>	
<input type="checkbox"/>	<a href="#">Crystal Multi-process Job</a>	3CRYSTAL	PSJob	(None)	(None)	<a href="#">Distribution</a>	
<input type="checkbox"/>	<a href="#">SQR Multi-process Job</a>	3SQR	PSJob	(None)	(None)	<a href="#">Distribution</a>	
<input type="checkbox"/>	<a href="#">6XREFWIN</a>	6SQR	PSJob	(None)	(None)	<a href="#">Distribution</a>	
<input type="checkbox"/>	Simple AE test program	AEMINITEST	Application Engine	Web	TXT	<a href="#">Distribution</a>	
<input type="checkbox"/>	<a href="#">All Process Types</a>	ALLTYPES	PSJob	(None)	(None)	<a href="#">Distribution</a>	
<input type="checkbox"/>	Field Cross Reference	XRFIELDS	Crystal	Web	PDF	<a href="#">Distribution</a>	
<input type="checkbox"/>	Export User Tables	EXPTUSR	Data Mover	Web	TXT	<a href="#">Distribution</a>	

Process Scheduler Request page

6. Select the row in the Process List grid containing the attributes *Field Cross Reference*,*XRFIELDS*, and *Crystal* options by clicking the Select check box next to each of these values in the Process List grid.
7. Click OK to generate a formatted display of this Crystal report in HTML.



## Appendix H

# Working With Records Used in the Mobile Synchronization Framework

This appendix provides an overview of records used in the Mobile Synchronization Framework, discusses how to configure records for the Mobile Synchronization Framework, and provides an example.

---

**Important!** PeopleSoft Mobile Agent is a deprecated product. These features exist for backward compatibility only.

---

---

## Understanding Mobile Synchronization Framework Records

Because of the architecture of the Mobile Synchronization Framework, you must consider additional settings when building applications that you intend to run on mobile or disconnected computers. These settings address updates to the application database that is running on the main network while mobile users have their computers in a disconnected state.

With the Mobile Synchronization Framework, the mobile user has a copy of the application tables running on their computers. If, for example, a system administrator adds a column to a table used in the mobile application, then the copy of this same table running on the disconnected computer is no longer current. When mobile users connect to the main network and attempt to synchronize the data on their machines with the data in the main application database, the synchronization operation will fail if the new column is a required (or 'NOT NULL') field because the application database expects a value to be inserted in the new column.

To ensure that the synchronization operation completes successfully, the system allows default values to be inserted in the new column.

---

**Note.** You can only specify default values in fields of the following types: Character, Number, and Signed Number.

---

---

## Configuring Records for the Mobile Synchronization Framework

This section discusses how to:

- Set record properties.
- Work with required fields and default values.

## Setting Record Properties

On the Use tab of the Record Properties dialog, a group box named Sync type (MSF) applies only to records used in mobile applications. The options in the Sync type (MSF) group box enable you to identify records as being used for mobile applications by selecting either of the options.

**Server —> User (Down Sync)** Select this option if the data on the mobile computer needs to be updated with the data on the main application database server. Selecting this option does not affect any behavior of the build process, such as inserting default values for required fields.

**User —> Server (Up Sync)** Select this option if the data on the main application database server needs to be updated by the data on the mobile computer.

If this option is selected and a Character, Number, or Signed Number column is set as required, during the build process the system automatically inserts the default values for these fields.

Also, if this option is selected the system does not allow you to set the following field types as required fields:

- Long Character
- Date
- Time
- Datetime

---

**Note.** You can select both options, depending on the application requirements.

---

### See Also

[Chapter 6, "Creating Record Definitions," Setting Record Properties, page 127](#)

## Working with Required Fields and Default Values

This section discusses how to:

- Create a required field.
- Specify default values for required fields.

### ***Creating a Required Field***

As with any record definition, you specify that a field is required on the Record Field Properties - Edits tab. Selecting the Required check box specifies that a particular field is 'NOT NULL'.

---

**Note.** By default, the script automatically sets character and number fields to NOT NULL regardless of whether the Required check box is checked.

---

The same is true for records used in mobile applications, except for records that have the User —> Server (Up Sync) option set. If set, you can only set Character, Number, and Signed Number fields to required. For all other field types, the system disables the Required check box.

### ***Specifying Default Values for Required Fields***

If the User —> Server (Up Sync) option is set in the record properties *and* the Required option is set in the record field properties, then during the build process the system inserts the following default values:

<b><i>Field Type</i></b>	<b><i>Default Value</i></b>
Character	'<space>'
Number	0 (zero)
Signed Number	0 (zero)

However, you can specify customer default values on the Use tab of the Record Field Properties dialog box. As with all other fields, you use the Default Value group box to specify custom options.

---

## **Example: Building Records for the Mobile Synchronization Framework**

This example illustrates the SQL generated for the build scripts for records that meet the following criteria:

- Used in mobile applications.
- Have the User —> Server (Up Sync) record property set.
- Contain Long, Numbered, or Signed Number fields with the record field property of Required set.

The following SQL examples create a table with four NOT NULL fields in the mobile record PS\_MSF\_DEMO. Two are Character fields (CUST\_ID and AA\_STATUS), one is a Number field (STD\_HOURS), and one is a Signed Number field (QTY\_ORDERED).

This sample SQL has no custom defaults specified on the Edits tab of the Record Field Properties dialog box:

```
CREATE TABLE PS_MSF_DEMO_1 (CUST_ID NCHAR(15) DEFAULT N' ' NOT NULL,
  STD_HOURS DECIMAL(6, 2) DEFAULT 0 NOT NULL,
  QTY_ORDERED DECIMAL(14, 4) DEFAULT 0 NOT NULL,
  ACTION_DT PDATE NULL,
  PM_VALID_TO_TIME PTIME NULL,
  APPROVEDTM PSDATETIME NULL,
  AA_STATUS NCHAR(1) DEFAULT N' ' NOT NULL,
  QE_TEST_DESCR NTEXT NULL)
```

This sample SQL contains custom defaults that are specified on the Edits tab of the Record Field Properties dialog box. For STD\_HOURS, the default is 8; for QTY\_ORDERED, the default is 1; and for AA\_STATUS, the default is A:

```
CREATE TABLE PS_MSF_DEMO_1 (CUST_ID NCHAR(15) DEFAULT N' ' NOT NULL,  
    STD_HOURS DECIMAL(6, 2) DEFAULT 8 NOT NULL,  
    QTY_ORDERED DECIMAL(14, 4) DEFAULT 1 NOT NULL,  
    ACTION_DT PDATE NULL,  
    PM_VALID_TO_TIME PTIME NULL,  
    APPROVEDTM PSDATETIME NULL,  
    AA_STATUS NCHAR(1) DEFAULT N'A' NOT NULL,  
    QE_TEST_DESCR NTEXT NULL)
```

## Appendix I

# Creating Custom Plug-In Files for the Rich Text Editor

Plug-ins enable you to add custom buttons to the rich text editor toolbar.

This chapter discusses how to:

- Create custom buttons.
- Use images with plug-ins.
- Add multilingual support.
- Add plug-ins to the configuration file.
- Specify the Plug-In File ID value.

### **See Also**

<http://docs.cksource.com/>

---

## Creating Custom Buttons

To create custom buttons:

1. Create a new HTML definition and save it with the prefix `PT_RTE_PLUGIN_`, for example, `PT_RTE_PLUGIN_MYPLUGIN`.

2. Use JavaScript to code the functionality of the button in the HTML definition. This is sample code for a button that will create a new page:

```
CKEDITOR.plugins.add( 'TestPlugin',
{
  init : function( editor )
  {

    editor.addCommand( 'Oracle_New_Page',
    {
      modes : { wysiwyg:1, source:1 },

      exec : function( editor )
      {
        editor.setData( editor.config.newpage_html );
        editor.focus();
      }
    });

    var iconpath= CKEDITOR.getUrl('skins/' + editor.config.skin + '/icons.png' );
    editor.ui.addButton( 'TestPlugin',
    {
      label : ''Oracle_New_Page'',
      command : ''Oracle_New_Page'',
      icon: iconpath,
      iconOffset : 38
    });
  }
});

CKEDITOR.config.newpage_html = '';
```

---

## Using Images with Plug-Ins

To use images with plug-ins:

1. Add the images to the database by creating new image definitions in PeopleSoft Application Designer.
2. Create an iScript method to download the images you need for the plug-in, for example:

```
Function IScript_ GetImages
  &ImageURL = %Response.GetImageURL( Image.XYZ );
  %Response.Write(&ImageURL);
End-Function;
```

3. Send an AJAX request from the plug-in to the iScript method.

The response is the image URL.

4. Use the image URL to display the image for the plug-in.

---

## Adding Multilingual Support

To add multilingual support to the button:

1. Add any Message Catalog entries necessary for multilingual support.

2. Create an iScript method to get the language-specific information from the Message Catalog. This is an example:

```
Function IScript_ GetData
  %Response.Write (MsgGetText(1234, 1, "Message not found.));
End-Function;
```

3. Send an AJAX request from the plug-in to the iScript method.

The response is the language-specific information.

4. Use the data as the new language information for the button.

---

## Adding Plug-Ins to the Configuration File

To add plug-ins to the configuration file:

1. Add the new toolbar button to the toolbar set in the configuration file.

This is an example of the code for the new Oracle\_New\_Page button:

```
CKEDITOR.config.toolbar =
[
  ['Preview','Print','-','Cut','Copy','Paste','-','Undo','Redo','->
  ','Find','Replace','Oracle_New_Page']
];
```

2. Add the plug-in that hosts the new button.

This is an example of the code:

```
CKEDITOR.plugins.addExternal( 'Oracle_New_Page', CKEDITOR.config.PluginPath );
CKEDITOR.config.plugins += ', 'Oracle_New_Page';

CKEDITOR.config.toolbar =
[
  ['Preview','Print','-','Cut','Copy','Paste','-','Undo','Redo','->
  ','Find','Replace','Oracle_New_Page']
];
```

If you are displaying the rich text editor in a limited space, you must group the editor controls using brackets [ ], as you see in this example:

```
config.toolbar =
[
  ['Maximize','Preview','Print'], ['Cut','Copy','Paste'], ['Undo','Redo'], =>
  ['Find','Replace'],
  ['HorizontalRule','Table','imageUPLOAD'], ['Link','Unlink','SpecialChar'],
  ['Format','Font','FontSize'], ['Bold','Italic','Underline','Strike'],
  ['JustifyLeft','JustifyCenter'], ['JustifyRight','JustifyBlock'],
  ['NumberedList','BulletedList'], ['Outdent','Indent'],
  ['TextColor','BGColor']
];
```

If you do not group the editor controls in this way, the application will not render the editor correctly.

---

## Specifying the Plug-In File ID Value

To specify the plug-in File ID value:

1. In PeopleSoft Application Designer, access the properties for the rich text-enabled long edit box.
2. Click the Options tab.
3. Select the name of the plug-in file from the Plugin File Id drop-down list box.
4. Save the page definition.



## Appendix J

# Configuring Rich Text-Enabled Fields to Store Accented Western European Characters

By default, when you enter accented Western European characters in a rich text-enabled field, the rich text editor encodes the accented characters as HTML entities before the system stores the data in the database. For example, the word *Französisch* is stored as `<p>Franz&ouml;sisch</p>`. As a result of being encoded as HTML entities, the characters (and affected data) appear as the user entered the data only when viewed in HTML, such as on a transaction page. When the data is viewed in a non-HTML format, such as reporting, searches, queries, or direct SQL, the characters do not appear as expected.

You can enable support for accented characters at these two levels:

- The individual rich text field.
- System-wide.

---

## Enabling Field-Level Support For Accented Characters

To enable support for accented characters at the individual field level:

1. Create a custom HTML definition for the specific field.

See [Chapter 11, "Creating Page Definitions," Creating Custom Configurations, page 359](#).

2. Add these two lines to the HTML in the definition:

```
CKEDITOR.config.entities_latin = false;  
CKEDITOR.config.entities_greek = false;
```

3. Save the HTML definition.

---

**Important!** You must use this naming convention: `PT_RTE_CFG_<name>`.

---

4. To use the new configuration, open the Long Edit Box Page Field Properties dialog box; select the Options tab; click the Configuration Settings Iddrop-down list box; and then select the HTML definition name from the list of values that appear.

See [Chapter 11, "Creating Page Definitions," Setting Page Field Properties for Controls, page 337](#).

5. Click the OK button and save the page definition.

## Sample HTML

The two relevant lines of code appear in bold in this sample HTML:

```

/*
Copyright (c) 2003-2009, CKSource - Frederico Knabben. All rights reserved.
For licensing, see LICENSE.html or http://ckeditor.com/license
*/

CKEDITOR.config.entities_latin = false;
CKEDITOR.config.entities_greek = false;
CKEDITOR.editorConfig = function( config )
{
    // Define changes to default configuration here. For example:
    // config.autoLanguage = false;
    // config.defaultLanguage = 'pt-br';
    config.skin='office2003';
    config.resize_enabled = false;
    config.removePlugins = 'elementspath';

    config.toolbar =
    [
        [ 'Maximize', 'Preview', 'Print', '-', 'Cut', 'Copy', 'Paste', '-', 'Undo' =>
        , 'Redo', '-', 'Find', 'Replace', '-', 'HorizontalRule', 'Table', 'imageUPLOAD' =>
        , 'Link', 'Unlink', 'SpecialChar' ],
        [ 'Format', 'Font', 'FontSize', '-', 'Bold', 'Italic', 'Underline', 'Strike' ],
        [ 'JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyBlock', '-' =>
        , 'NumberedList', 'BulletedList', 'Outdent', 'Indent', '-', 'TextColor', 'BGColor' ]
    ];

    config keystrokes =
    [
        [ CKEDITOR.ALT + 121 /*F10*/, 'toolbarFocus' ],
        [ CKEDITOR.ALT + 122 /*F11*/, 'elementsPathFocus' ],
        [ CKEDITOR.SHIFT + 121 /*F10*/, 'contextMenu' ],
        [ CKEDITOR.CTRL + 90 /*Z*/, 'undo' ],
        [ CKEDITOR.CTRL + 89 /*Y*/, 'redo' ],
        [ CKEDITOR.CTRL + CKEDITOR.SHIFT + 90 /*Z*/, 'redo' ],
        [ CKEDITOR.CTRL + 76 /*L*/, 'link' ],
        [ CKEDITOR.CTRL + CKEDITOR.SHIFT + 76 /*L*/, 'unlink' ],
        [ CKEDITOR.CTRL + 66 /*B*/, 'bold' ],
        [ CKEDITOR.CTRL + 73 /*I*/, 'italic' ],
        [ CKEDITOR.CTRL + 85 /*U*/, 'underline' ],
        [ CKEDITOR.CTRL + CKEDITOR.ALT + 13 /*ENTER*/, 'maximize' ],
        [ CKEDITOR.CTRL + 77 /*M*/, 'imageUPLOAD' ]
    ];
};

```

---

## Enabling System-Wide Support For Accented Characters

To enable system-wide support for accented characters in rich text fields:

1. Navigate to *PS\_HOME\webserv\your web server name\applications\peoplesoft\PORTAL.war\sitename\ckeditor*.

2. Open the `config_psft.js` file, which is located in the `ckeditor` folder.
3. Add these two lines to the beginning of the configuration file:  

```
CKEDITOR.config.entities_latin = false;  
CKEDITOR.config.entities_greek = false;
```
4. Save the configuration file.
5. Restart the webserver.

---

## Sample `config_psft.js` File

This is a sample configuration file:

```
/*  
Copyright (c) 2003-2009, CKSource - Frederico Knabben. All rights reserved.  
For licensing, see LICENSE.html or http://ckeditor.com/license  
*/  
  
CKEDITOR.config.entities_latin = false;  
CKEDITOR.config.entities_greek = false;  
var ImageUpload = CKEDITOR.getUrl('imageupload.js');  
CKEDITOR.plugins.addExternal( 'imageUPLOAD', ImageUpload );  
CKEDITOR.config.plugins += ',imageUPLOAD';  
CKEDITOR.config.plugins = CKEDITOR.config.plugins.replace(',image', ' ');  
CKEDITOR.config.menu_groups = CKEDITOR.config.menu_groups.replace(',image', ' ');  
CKEDITOR.config.menu_groups = CKEDITOR.config.menu_groups.replace(',imagebutton', ' ');  
CKEDITOR.config.menu_groups += ',imageUPLOAD';  
CKEDITOR.config.scayt_autoStartup = false;  
CKEDITOR.on( 'instanceReady', function( evt )  
{  
  
    function noBubbling( evt ) { evt.data.stopPropagation(); }  
    // Stop keystroke event bubbling from editor chrome.  
    var container = evt.editor.container;  
    container.on( 'keydown', noBubbling );  
    container.on( 'keypress', noBubbling );  
    container.on( 'keyup', noBubbling );  
});
```



## Appendix K

# Creating Custom SmartNavigation Pages

While the appearance of SmartNavigation pages is unlike the standard PeopleSoft application page, the process and the definitions that you use to create both are the same. Field, record, page, and component definitions comprise SmartNavigation pages, just as those definitions comprise standard application pages. The primary difference is that you use a specific subset of definitions and a specific configuration for all SmartNavigation pages because of the manner in which the processing logic renders the chart and populates it with data.

This chapter provides an overview of the elements of SmartNavigation pages.

---

## Understanding the Elements of SmartNavigation Pages

The standard PeopleTools SmartNavigation component, PT\_ABN\_ORGCHART, is the template for all SmartNavigation pages. It contains specific, purposefully ordered page and record definitions that work in conjunction with the processing logic of PeopleSoft applications. The definitions that comprise the delivered component serve as the foundation not only of the default SmartNavigation page, but also of any SmartNavigation page that you create for your own purposes.

Because the processing logic expects SmartNavigation pages to have a specific component processing structure, you should copy the existing definitions, modify the copied definitions, and add functionality to the page with fields, page controls, and PeopleCode that accommodate the following technical requirements in addition to your business rules.

You should create a new definition for SmartNavigation pages according to the following process:

1. Clone the PT\_ABNORGND\_SBR, PT\_ABNURL\_SBR, PT\_ABNNDDTL\_SBR and PTORGBOXFLD\_SBR subrecords and the accompanying PeopleCode.

The PT\_ABNORGND\_SBR subrecord contains the fields and PeopleCode that populate the nodes with the correct detail information and application data as your users navigate the SmartNavigation page.

2. Modify your record PeopleCode as necessary to reference your definitions; use the record PeopleCode in the PT\_ABNORGND\_SBR subrecord as the model for any PeopleCode that you use in your record definition.
3. Clone the PT\_ABN\_CHART\_ND record definition and its accompanying PeopleCode.
4. Remove the original subrecords and insert the clones of the PT\_ABNORGND\_SBR, PT\_ABNURL\_SBR, PT\_ABNNDDTL\_SBR and PTORGBOXFLD\_SBR subrecords in that order.
5. Modify your record PeopleCode as necessary to reference your definitions; use the record PeopleCode in the PT\_ABNORGND\_SBR subrecord definition as the model for any PeopleCode that you use in your record definition.

6. If you need to add new fields, add them to the bottom of the field list.

---

**Important!** Do not add fields to the required subrecords or place fields between these subrecords when you create your record definition.

---

7. Clone the PT\_ABN\_ORGCHART page definition and its accompanying PeopleCode.

This page contains the fields and PeopleCode that define the chart on the SmartNavigation page.

8. Modify the page field controls so that they reference the cloned record definition that you created in Step 3.
9. Modify your page PeopleCode as necessary to reference your definitions; use the page PeopleCode in the PT\_ABN\_ORGCHART page definition as the model for any PeopleCode that you use in your page definition.
10. Select the Order tab and verify that the order and levels of the page fields are identical to the order and levels of the page fields in the default PT\_ABN\_ORGCHART page definition.
11. Add new page fields to the bottom of the page definition and make sure that they appear last in the page field order.
12. Clone the PT\_ABN\_ORGCHART component definition and its accompanying PeopleCode.
13. Remove the existing page and insert the page you cloned in Step 7.
14. Modify your component PeopleCode as necessary to reference your definitions; use the component PeopleCode in the PT\_ABN\_ORGCHART component definition as the model for any PeopleCode that you use in your component definition.
15. In the browser, configure your SmartNavigation folder to use your new SmartNavigation page.
16. Deselect the Default Chart Navigation Page check box and enter the name of your SmartNavigation page in the Folder Navigation Object Name field.

This table provides references to resources where you can find details that are specific to each step in the process:

<b>Step</b>	<b>PeopleBook Reference</b>
1, 2, 3, 4, 5, 6	See <a href="#">Chapter 6, "Creating Record Definitions," page 115</a> ; <a href="#">Chapter 6, "Creating Record Definitions," Inserting Subrecords Into Records, page 126</a> and <a href="#">Chapter 6, "Creating Record Definitions," Editing PeopleCode in Subrecords, page 126</a> .
7, 8, 9, 10, 11	See <a href="#">Chapter 11, "Creating Page Definitions," Cloning Page Definitions, page 329</a> ; <a href="#">Chapter 11, "Creating Page Definitions," Adding Page Controls, page 329</a> ; <a href="#">Chapter 11, "Creating Page Definitions," Setting Record Properties, page 338</a> ; <a href="#">Chapter 11, "Creating Page Definitions," Changing Control Order Using the Order Tab, page 369</a> and <a href="#">Chapter 11, "Creating Page Definitions," Accessing PeopleCode in Page Definitions, page 373</a> .
12, 13, 14	See <a href="#">Chapter 12, "Creating Component Definitions," Adding Pages to Components, page 397</a> and <a href="#">Chapter 12, "Creating Component Definitions," Structure Tab, page 396</a> .

Step	PeopleBook Reference
15, 16	See <i>PeopleTools 8.52: PeopleTools Portal Technologies</i> , "Administering Portals," Defining SmartNavigation Folders.





## Appendix L

# Configuring Embedded Help to Display UPK Content

As part of the embedded Help functionality, you can include links to Oracle UPK modules and topics. This appendix describes how to:

- Modify the UPK Player package link.
- Add UPK-based URLs to the application.
- Embed UPK-based URLs in messages.
- Add embedded Help to page controls.

---

**Note.** In this appendix, the word *topic* is used to represent both topics and modules unless specified otherwise.

Your UPK content should already be installed on the web server and available to your users.

---

---

## Modifying the UPK Player Package Link

To modify the UPK player package link:

1. Contact your UPK system administrator to obtain the player package link (URL). It should look similar to this:

*[http://upkservername/upk/content\\_subject/toc.html](http://upkservername/upk/content_subject/toc.html)*

2. Change the file name toc.html to kp.html so that the URL looks similar to this:

*[http://upkservername/upk/content\\_subject/kp.html](http://upkservername/upk/content_subject/kp.html)*

The kp.html file lists your UPK modules and topics. Pointing directly to the kp.html file, which is located in the player package folder, enables you to manually link to a specific mode for a topic.

3. Paste the link into your browser to access the file.
4. Find the topic that you want to use in embedded help, and then right-click the Concept or Do It link associated with the topic.

If you are creating task oriented related content, right-click the Do It link, otherwise click the Concept link.

- From the pop-up menu, select Copy Shortcut or Copy Link Location. You will need the URL in the next topic, *Adding UPK-Based URLs to the Application*.

As an alternative, select Properties and then copy the URL that you see on the dialog box.

- For conceptual information, the URL should look similar to this:

*<http://upkserver/upk/hcmam91/tpc/1d1a9fb4-aa18-407a-b50c-f5e0e65e4bb8/index.html>*

- For task oriented information, the URL should look similar to this:

*[http://upkserver/upk/peopletools/dhtml\\_kp.html?Guid=69f0376e-70fc-417c-a43cecd6afb72b29&Mode=D&Back](http://upkserver/upk/peopletools/dhtml_kp.html?Guid=69f0376e-70fc-417c-a43cecd6afb72b29&Mode=D&Back)*

---

**Note.** The URL for each UPK mode ends differently. Look for &Mode= near the end of the URL.

---

This table lists the different URL endings and the corresponding UPK mode for each ending:

<b>URL Ending</b>	<b>UPK Mode</b>
&Mode=D	Do It
&Mode=K	Know It
&Mode=T	Try It
&Mode=S	See It
&Mode=P	Print It
Index.html	Concept

---

## Adding UPK-Based URLs to the Application

To add UPK-based URLs to the application:

- Select Main Menu, PeopleTools, Utilities, Administration, URLs.
- Add a new value, such as UPK\_QRY\_1.

This is the URL identifier. You will need this later.

- Click the Add button.
- Enter a description.
- Paste the URL the you copied in the previous topic into the URL field.
- Remove *&Back* from the end of the URL so that the URL ends in the mode, for example, *&Mode=D*.
- Save the URL.

---

## Embedding UPK-Based URLs in Messages

To embed UPK-based URLs in messages:

1. Select Administration from the drop-down menu, and then select Message Catalog.
2. Click Add a New Value.
3. Enter 21000. (You want to enter a value that isn't reserved by PeopleTools or PeopleSoft applications.)
4. Enter a description.
5. Enter message text.

---

**Note.** The message text is the link text that appears in the embedded help pop up. A phrase such as, "Show me" or another generic phrase works well.

---

6. Enter a description. In the description, include this code:

```
%PBURLTBL(URL_ID, setnbr, mnbr)
```

7. Replace *URL\_ID* with the URL identifier that you added earlier.
8. Replace *setnbr* with the set number of your message.
9. Replace *mnbr* with the message number of your message.

The following example would access the UPK\_QRY\_1 URL and display message set 3, message number six:

```
%PBURLTBL(UPK_QRY_1, 3, 61)
```

10. Save the message.

---

## Adding Embedded Help to Page Controls

To add embedded Help to page controls:

1. Open the page definition.
2. Double-click the scroll area or grid to access the properties dialog box.
3. Click the Label tab.
4. In the Embedded Help Popup section, select the Message Catalog Popup option.
5. Enter the Message Set/Number values of the message that you want to appear in the embedded help pop-up page.

The message should appear in the Text field.

6. Verify that the message in the Text field is correct.

7. Click the OK button.
8. Save the page definition.
9. In the browser, access the page and test the embedded help.

# Index

## Symbols

%EditTable function 203  
%Market system variable 402

## A

access keys, setting 401  
access profiles  
    locking/unlocking definitions 563  
    understanding 44  
    using deferred mode 420  
activities, workflow 28  
Add Translate Table Value dialog box 94  
Align  
    page control labels 325  
    page controls 325  
alignment  
    controlling via style classes 455  
    setting button label 259  
    setting in style sheets 458  
    setting label 326  
    setting page control 326, 341, 371  
    setting page control label 371  
    setting page label 347  
    setting static text 221  
alternate search keys  
    setting for records 136  
    understanding 134  
    using in indexes 176  
Alter process  
    confirming table alterations 192  
    creating/running alter scripts 184  
    creating temporary tables 191  
    determining tables to alter after field changes 187  
    forcing without table changes 190  
    handling data conversions 182  
    performing Alter in Place operations 190  
    renaming table fields 82  
    running 188  
    setting alter options 181, 188  
    setting column options 189  
    setting custom options 190  
    understanding 181  
    when to run 187  
analytic model 28  
application classes 28  
application data tables  
    creating/understanding 97  
    synchronizing with records 164  
    using the Alter process 187  
Application Designer  
    configuring 41  
    hiding the interface 13, 161  
    implementing 3  
    menus 14  
    setting undo/redo layers 22  
    setting Word Wrap for editors 22

    starting 11, 532  
    understanding 3, 9  
    window  
        *See Also* Application Designer window  
Application Designer window  
    definition workspace  
        *See Also* definition workspace  
    navigating 14  
    output window *See Also* output window  
    project workspace  
        *See Also* project workspace  
    using the components 14  
    using the toolbar 20  
application development  
    eight steps 10  
    planning records, control tables, and tablesets 97  
Application Engine programs  
    import *See Also* import program  
    understanding 28  
    using temporary tables 118  
application packages  
    accessing associated PeopleCode programs 26  
    Application Packages editor 26  
    understanding 28  
Applications and Fields report 632  
approval rule sets 28  
auditing  
    creating audit records 130  
    naming audit records 124, 128  
    setting field audit options 137  
    using record-level audits 128  
autocomplete  
    field configuration 139  
Auto Size Page 325

## B

backgrounds  
    changing colors for classic tabs 611  
    changing colors for tabs 612  
    pages, overriding style sheets for 385  
    pages, selecting backgrounds for 244  
    referencing image URLs 464  
    selecting for edit boxes 341  
    selecting for grids 273, 297  
    style classes, setting attributes for 461, 462  
    style classes, using 455  
    style classes, using default 591  
Body Area Row Action Button Properties dialog box 274, 298  
borders  
    displaying for grids 274, 298  
    hiding for frames 213  
    setting for group boxes 216  
    setting for page reports 388  
    setting for record definition reports 150  
    style classes, setting attributes for 455, 464  
    style classes, using default 591

- browsers
    - adjusting time fields for time zones 72, 101
    - changing default 377
    - configuring options 48
    - defining tableset controls 112
    - displaying fonts 457
    - editing/viewing DDL 197
    - HTML tables and performance 213, 373
    - indicating link status to users 468
    - opening for component search pages 411
    - ordering tabs 368
    - overriding date field formats 72
    - setting edit box colors 341
    - setting the base URI 23
    - sizing page controls 333, 341
    - spacing page controls 370
    - supporting buttons 259
    - supporting calendar prompts 321
    - supporting images 489, 493
    - supporting style class attributes 455
    - using WBMP images 488
    - View in Browser function
      - See Also* View in Browser function
    - viewing HTML for pages 379
    - viewing pages 375
    - wrapping long words 351
  - buffers
    - adding/deleting rows 204, 304
    - controlling scrollable data 279
    - displaying the buffer size for file layouts 514
    - setting undo/redo layers 22
    - understanding component buffers 394
  - Build dialog box 157, 159
  - Build menu 17, 157
  - Build process
    - Alter process *See Also* Alter process
    - building script files 158
    - Build Settings dialog box
      - See Also* Build Settings dialog box
    - configuring 160
    - creating triggers 192
    - errors, navigating to 171
    - errors for project builds 161
    - granting build authority 156
    - indexes, creating 163, 175, 177
    - indexes, setting create options for 167
    - logs, setting options for build 167
    - logs (sample), output build 169
    - monitoring 158
    - running 159
    - selecting the scope 159
    - setting options 160
    - setting parameters 17
    - setting script options 171
    - table builds, confirming 175
    - tables, building SQL 11
    - tables, creating 163, 164
    - tables, setting create options for 165
    - understanding 50, 155
    - using the Build menu 157
    - viewing messages 21
    - views, creating 192
    - views, setting create options for 166
    - views, setting the build order for dynamic 152
  - Build Settings dialog box
    - opening 160
    - setting alter options 188
    - setting create options 165
    - setting logging options 167
    - setting script options 171
    - understanding 157
  - bulk operations
    - deletions 629
    - insertions 625
    - modifications 627
    - performing in PeopleSoft financial applications 625
    - understanding 625
  - Business Interlink 28
  - business processes 28
  - buttons
    - activating via the Enter key 262
    - adding for grids 255
    - adding for scroll areas 255
    - adding for scroll bars 255
    - adding instant messaging 254
    - adding scroll action buttons 278
    - associating with PeopleCode commands 254
    - calendar prompt *See Also* calendar prompts
    - enabling for display-only access 261
    - hover text, setting 263
    - HTML entries for 378
    - inserting 251, 327
    - jumping to page anchors 254
    - labels, setting images for 259
    - labels, setting text for 257
    - prompt/lookup *See Also* prompt buttons
    - running processes 254
    - secondary pages, accessing 255
    - secondary pages, associating primary pages with 245
    - setting as external/internal links 252
    - setting destination types 252
    - understanding 211, 250
    - using default style classes 591
- ## C
- calendar prompts
    - button image for 618
    - understanding 321
  - catalog
    - image 488
  - Change Control
    - deleting page definitions 570
  - Change Control Change Control locking
    - understanding 551
  - Change Control feature
    - appointing an administrator 563
    - enabling/disabling 563
    - history *See Also* Change Control history
    - implementation considerations 555
    - locking *See Also* Change Control locking
    - reporting information 556
    - setting up 18, 562
    - stamping *See Also* Change Control stamping
    - supported definition types 561
    - understanding 551
    - understanding security 562
    - using distributed development environments 565
    - using multiple databases for development 565

- using projects 564
- Change Control history
  - enabling 564
  - implementation considerations 555
  - inserting/prompting automatically 553
  - inserting definition comments 568
  - opening entries 571
  - reporting 562
  - understanding 553
  - upgrading applications 554
  - using the PSCHGCTLHIST table 556
  - viewing 570
- Change Control locking
  - definitions, automatically locking 567
  - definitions, deleting 554
  - definitions, identifying locked 551
  - definitions, locking 566
  - definitions, locking all 564, 567
  - definitions, viewing locked 568
  - enabling 564
  - entering comments 553
  - implementation considerations 555
  - projects 552
  - understanding upgrades 552
  - understanding version control 552
  - using the PSCHGCTLLOCK table 556
- Change Control stamping
  - understanding 554
- change projects
  - file type codes, understanding 44
  - understanding 59
- Change Record Indexes dialog box 178
- change sets 564
- Change Space dialog box 196
- channels 29
- character fields
  - attributes, setting 67
  - attributes for the Alter process, setting 181
  - creating 65
  - Japanese 68
  - long *See Also* long character fields
  - understanding 65, 67
- characters
  - applying uppercase automatically 91
  - character fields *See Also* character fields
  - end of file (EOF) 527
  - interpreting backslashes as escape characters 515
  - long character fields
    - See Also* long character fields
  - naming definitions 33
  - naming records 123
  - replacing blank edit box spaces with fill
    - characters 343
  - using lexical, literal and meta 83, 90
  - using Smart Punctuation 90
- ChartFields
  - setting styles 594
  - understanding 625
- charts
  - ChartFields *See Also* ChartFields
  - drawing chart placeholders 208
  - inserting 327
  - using 212
- check boxes
  - associating with records 222
  - enable when page is display only 351
  - HTML entries for 378
  - inserting 327
  - labels, enabling/disabling colons in 348
  - labels, using default 221
  - setting properties 221
  - understanding 209
  - using default style classes 591
  - using in deferred mode 203
- child tables 132
- Class dialog box
  - setting backgrounds 461
  - setting border attributes 464
  - setting fonts 456
  - setting margin attributes 465
  - setting miscellaneous attributes 467
  - setting spacing and alignment 458
- colors
  - changing colors for classic tabs 611, 612
  - changing colors for tabs 481, 612
  - modifying system colors 43
  - selecting for edit boxes 341
  - setting for borders 465
  - setting for fonts 458
  - setting for grids 305
  - setting for page controls 341
  - style classes, setting attributes for 461
  - style classes, using 455
  - style classes, using default 591
- columns, grid
  - aligning labels 347
  - deleting 284
  - enabling View All 307
  - freezing 288
  - inserting/manipulating 283
  - moving 285
  - ordering 282
  - setting heading properties 299
  - setting options in the Alter process 189
  - setting properties 285
  - showing when cells hidden 306
  - sizing 122, 285
  - sorting 122, 303
- command line
  - building projects 161
  - parameters
    - See Also* command line parameters
  - starting Application Designer 12
- command line parameters
  - building projects 534
  - comparing definitions 542
  - copying definitions 536
  - project builds 161
  - referencing selections 579
  - starting Application Designer 532
  - starting Data Mover 532
  - syntax 12, 531
  - understanding 531
- comments
  - enabling the prompt for definitions 569
  - including/excluding alter comments 172
  - inserting for definitions 568
- ComponentChanged function 262
- component definition window 395
- component interfaces
  - deleting 35
  - understanding 28
- Component Properties dialog box
  - opening 403
  - setting general properties 403

- setting Internet properties 406
  - setting use properties 404
  - components
    - adding to a portal registry 437
    - adding to permission lists 438
    - assigning to portals 436
    - ComponentChanged function 262
    - Component Properties dialog box
      - See Also* Component Properties dialog box
    - context search records, setting 405
    - customizing the display 412
    - defining 395, 397
    - defining pagebars 411
    - definitions, creating 11
    - definitions, opening 397
    - definitions, opening record 396
    - deleting 35
    - displaying/updating record rows 405
    - displaying toolbar actions 409
    - enabling the Expert Entry feature 410
    - enabling the Expert Entry feature via
      - PeopleTools Security 421
    - including the Refresh button 420
    - interfaces *See Also* component interfaces
    - links, displaying 410
    - links, selecting components for internal 256
    - menu navigation, including components for 406
    - menus, adding components to 427, 437
    - menus, creating custom 424
    - menus, finding components that reference 403
    - opening attached PeopleCode 396
    - ordering within folders 440
    - page attributes, setting 399
    - page definitions, opening 395
    - page items/attributes, viewing 395
    - page level information, viewing 396
    - pages, adding 397
    - pages, copying/moving between components 399
    - pages, enabling deferred mode for 401
    - pages, hiding 400
    - pages, naming 400
    - pages, navigating 394, 401, 410
    - pages, reordering 398
    - pages based on work records, adding 415
    - processing modes, setting 410
    - processing modes based on users, setting 420
    - registering 11, 436
    - registering in multiple portals 442
    - search page history, enabling check boxes 408
    - search page options, setting 408
    - search pages, using instructional text for 409
    - search records, overriding 419
    - search records, setting 405
    - setting access keys 401
    - setting market-specific 402
    - setting properties
      - See Also* Component Properties dialog box
    - setting the three-tier execution location 406
    - spell check, enabling mandatory 406
    - spell check errors, understanding 46
    - tabs, displaying 410
    - tabs, labelling 400
    - understanding 28, 393
    - understanding buffers 394
    - validating 21
    - viewing record/role information 396
    - viewing update history 404
  - configuring embedded help 307
  - configuring persistent searches 141
  - Consolidate Images dialog box 494
  - content references
    - assigning names 436
    - assigning to portals 436
    - creating for components 429, 437
    - creating for pages 429
    - labelling 440
    - understanding 435
    - using default style classes 591
  - control tables
    - prompt tables *See Also* prompt tables
    - record groups *See Also* record groups
    - sharing values 102, 103
    - tablesets *See Also* tablesets
    - understanding 102
  - Convert Images dialog box 493
  - Convert Scroll to Grid function 279
  - Create Table process 163
  - cross-reference reports *See* Crystal reports
  - Crystal reports
    - running 633
    - understanding 631
  - CSV files
    - converting tabs to spaces 514
    - field padding 520
    - importing fields 522
    - inserting segments/fields 528
    - interpreting backslashes as escape characters 515
    - interpreting backslash quotes (\") 515
    - interpreting double quotes (") 515
    - parsing file fields 514
    - parsing support 505
    - previewing input data 522
    - selecting field qualifiers 520
    - trimming field spaces 521
    - troubleshooting input data 524
    - understanding 502, 504
  - currencies
    - displaying symbols 342
    - multicurrency fields 216, 221, 350
  - cursors
    - selecting page control groups 332
    - style classes, setting attributes for 468
    - style classes, using 455
    - using the page control toolbar 331
  - custom indexes
    - adding 179
    - understanding 176
- ## D
- database
    - validating for WSRP 18
  - databases
    - Change Control, using 565
    - Change Control stamping 554
    - command line parameters 531
    - comparing via command line parameters 542
    - copying between databases via command line parameters 536



- indexes *See Also* indexes
- normalized relational 98
- saving script files 172
- selecting types 579
- synchronizing definitions 50
- tables for PeopleTools applications 97
- triggers *See Also* triggers
- understanding forms 99
- using master development databases 565
- viewing space allocation 18
- Data Definition Language (DDL) *See* DDL
- data interchange *See* EDI
- Data Mover 532
- date fields
  - creating 65
  - enabling four-digit century entries 342
  - field length in file layouts 511
  - setting attributes 72
  - understanding 65, 72
- dates
  - date fields *See Also* date fields
  - DateTime fields *See Also* DateTime fields
  - effective *See Also* effective dates
  - selecting formats/separators for file fields 518
  - understanding reasonable 146
  - using the ReadRowset method 512
- Datetime fields
  - allowable formats 511
- DateTime fields
  - creating 65
  - displaying time zones 343
  - field length in file layouts 511
  - setting attributes 72
  - understanding 65, 72
- DB2 UDB
  - altering tables – impact on views 186
  - functional indexes 177, 575
  - maximum image field size 73
  - performing Alter in Place operations 191
  - saving script files 172
- DDL
  - administering 195
  - editing 180
  - understanding 156
  - understanding the Build process 155
  - using record DDL 195
  - viewing parameters for creating tables, indexes, and tablespaces 197
- DDL Model Defaults page 197
- Debug menu 17
- decimal precision
  - inserting decimals automatically for page controls 343, 344
  - meta symbols 84
  - selecting for file fields 519
  - setting for field definitions 119
  - setting for numeric fields 71
  - using CSV files 505
  - using FIXED files 511, 514
  - using XML files 507
- deferred mode
  - adding multiple rows 204
  - enabling for page controls 386
  - enabling for pages 386, 401
  - setting fields with edit tables 202
  - setting for components 410
  - setting for grids/scroll areas 303
  - triggering server processing 200
  - understanding 200
  - understanding performance issues 373
  - using check boxes and radio buttons 203
  - using drop-down list boxes 203
  - using error/warning messages 205
  - using on prompt tables 203
  - using PeopleCode 201
  - using related displays 203
  - using the Expert Entry feature 420
- deferred processing mode *See* deferred mode
- definition properties dialog box 31
- definitions
  - access profiles *See Also* access profiles
  - Change Control, definitions supported by 561
  - Change Control history, viewing 570
  - Change Control implementation considerations 555
  - Change Control stamping
    - See Also* Change Control stamping
  - closing 34
  - comments, enabling the prompt for 569
  - comments, entering 554
  - comments, inserting 568
  - component *See Also* components
  - constructing SQL 26
  - creating 33
  - definition workspace
    - See Also* definition workspace
  - deleting 35
  - drag-and-dropping 39
  - field *See Also* fields
  - field formats 42
  - file type code *See Also* file type codes
  - locked, viewing 568
  - locking 551, 566
  - locking all 564, 567
  - locking automatically 567
  - naming 33, 34
  - opening 15, 30
  - page *See Also* pages
  - PeopleCode, parent definitions for 24
  - PeopleCode, viewing 24
  - PeopleCode, viewing characteristics 16
  - PeopleCode definition types/IDs 560
  - PeopleTools definition types/IDs 557
  - performing bulk operations 19
  - projects, accessing definitions for 52
  - projects, inserting into 17, 55, 60
  - projects, modifying definitions for 54
  - projects, removing from 57
  - projects, using 51
  - properties, editing 15, 31
  - properties, viewing 31
  - record *See Also* records
  - references, finding
    - See Also* Find Definition References
  - process
  - references, understanding 37
  - references, viewing 16
  - related definitions, finding
    - See Also* Find Definition References
  - process
  - related definitions, understanding 37
  - reports for definitions, viewing 15
  - saving 33, 60
  - selecting types 580

- setting default owner IDs 23
- styles *See Also* styles
- synchronizing database definitions 50
- system colors *See Also* colors
- understanding the Build process 155
- understanding types 27
- upgrade, displaying definitions available for 53
- using definition keys 556
- viewing 15, 31
- definition workspace
  - cascading/closing/grouping/tiling definitions 19
  - opening definitions 30
  - understanding 21
  - using the multiple document interface (MDI) 38
- Delete Definition dialog box 35, 82
- development database 565
- Development view
  - identifying locked definitions 551
  - understanding 52
  - using 52
  - viewing upgrade-only definitions 57
- device independent bitmap (DIB) images 493
- DIB images 493
- DoModal function 211
- drag-and-drop
  - adding file fields to file layouts 510
  - adding file records to file layouts 509
  - copying/moving controls within/between pages 335
  - dragging field definitions from records to pages 330
  - dragging fields from the project workspace to a page 330
  - dragging pages into components 398
  - dragging record definitions onto pages 330
  - inserting columns into grids 283
  - inserting fields into records 125
  - moving fields between records 147
  - moving fields within records 146
  - moving grid columns 285
  - moving HTML areas 248
  - moving page control labels 335
  - ordering page controls via the order list 369
  - reordering component pages 398
  - using 39
- drop-down list boxes
  - default base styles 352
  - default style classes 591
  - HTML entries for 378
  - inserting 327
  - persist grid column in menu 352
  - positioning labels 347
  - setting properties 222
  - understanding 210
  - using 320
  - using deferred mode 203
  - using display-only 352
  - using the Expert Entry feature 421
  - wrapping long words 351
- duplicate order keys
  - ordering in records 134
  - setting for records 135
  - using in records 133
- dynamic views
  - creating select statements 152

- defining records as 118
- naming records as 123
- setting the build order 152
- using cross-platform meta-SQL 194

## E

- EDI
  - applying file layouts to 502
  - EDI Manager *See Also* EDI Manager
  - performing 521
  - using segments 509
- EDI Manager
  - CSV file properties 504
  - FIXED file properties 503
- edit boxes
  - as mouse over pop-up fields 225
  - default base styles 352
  - default style classes 591
  - enabling 1000 separators 342
  - enabling spell check 343
  - HTML entries for 378
  - inserting 327
  - labels, applying 225
  - labels, enabling/disabling colons in 348
  - labels, positioning 347
  - long *See Also* long edit boxes
  - persist grid column in menu 352
  - related control field 352
  - replacing blank spaces with fill characters 343
  - sizing 341
  - understanding 210
  - using display-only 225, 352
  - wrapping long words 351
- Edit Box Properties dialog box
  - setting general properties 354
  - setting label properties 344
  - setting record properties 338
  - setting use properties 348
- Edit menu 15, 324
- editors *See Also* SQL editor
- Edits Display mode 121
- effective dates
  - level-based controls 318
  - translate table values, adding 95
  - translate table values, understanding 94
  - understanding 100
  - using drop-down list boxes 203
  - using in components 405
- electronic data interchange (EDI) *See* EDI
- embedded help
  - adding to grids 310
  - adding to scroll areas 310
  - configuring 307
  - on group boxes 311
  - setting for group boxes 215
- embedded help pop-up properties 272, 296
- end of file (EOF) characters 527
- Enter key, activating links/buttons via 262
- EOF characters 527
- errors
  - displaying in deferred mode 205
  - handling for pages/page-controls 386
  - handling for project builds 161
  - logging during alter processes 161

- setting build log options 167
- setting error message styles 594
- spell check results 45
- troubleshooting file layout input data 524
- viewing for PeopleCode 21
- events
  - naming records with event fields 124
  - triggering server processing 200
  - viewing for field definitions 25, 121
- Excel, Microsoft *See* MS Excel
- Expert Entry feature
  - enabling 410, 420
  - enabling via PeopleTools Security 421
  - understanding 420
  - understanding limitations 421
  - using the Refresh button 420

## F

- Field Display mode 119
- field labels 335
- Field Listing report 632
- fields
  - applying uppercase automatically 91
  - character *See Also* character fields
  - containing PeopleCode 121
  - date *See Also* date fields
  - datetime *See Also* DateTime fields
  - definitions, changing 80
  - definitions, creating 10, 64
  - definitions, deleting 82
  - definitions, dragging 330
  - definitions, opening 78
  - definitions, printing 83
  - definitions, renaming 81
  - definitions, setting attributes for 65
  - definitions, understanding 63
  - definitions, viewing 79, 119
  - file fields *See Also* file fields
  - format families, changing 86
  - formats, adding display formats to stored 89
  - formats, adding stored 89
  - formats, deleting stored 89
  - formats, editing 88
  - formats, testing 91
  - formats, understanding 42
  - formats, updating display 89
  - formats and notation, understanding 83
  - image *See Also* image fields
  - ImageReference
    - See Also* ImageReference fields
  - key *See Also* key fields
  - label IDs, setting 138
  - labels, adding 76
  - labels, deleting 77
  - labels, selecting record field 77
  - labels, using multiple 75
  - long character *See Also* long character fields
  - numeric *See Also* numeric fields
  - page controls, setting properties for 337
  - page field attributes, bulk changes for 629
  - page fields, bulk modification of 628
  - pages, bulk insertion into 626, 627
  - pages, deleting from 148
  - pages, dragging fields from the project workspace to 330
  - record field properties, understanding 131
  - record fields, bulk modification of 628
  - record field use properties, setting 134
  - records, adding SetIDs to 109
  - records, bulk insertion into 626, 627
  - records, deleting from 147
  - records, inserting into 125
  - records, moving between 147
  - records, moving within 146
  - records, understanding fields in 63, 125
  - records, using audit fields in 130
  - records, using search fields in 137
  - refreshing 200
  - related display *See Also* related display
  - reordering 122
  - report, Applications and Fields 632
  - report, Field Listing 632
  - report, Fields and Pages 631
  - report, Fields and Records 148, 631
  - report, Fields Referenced by PeopleCode Programs 148, 632
  - report, PeopleCode Programs and Field References 632
  - report, Records and Fields 631
  - set control *See Also* set control fields
  - setting as required 144
  - setting audit options 137
  - setting autocomplete 139
  - setting default values 139
  - setting edit properties 143
  - setting related date fields 140
  - setting the maximum display length 88
  - setting time zone fields 140
  - table data, converting 182
  - tables, enabling prompt 145
  - tables, enabling translate 145
  - tables to alter after field changes,
    - determining 187
  - testing for reasonable dates 146
  - time *See Also* time fields
  - types, changing 80
  - types, understanding 65
  - understanding 29
  - understanding derived/work 322
  - understanding model fields 626
  - understanding spell check errors 47
  - using smart punctuation 90
  - viewing editing options 121
  - viewing key-related characteristics and default values 120
- Fields and Pages report 631
- Fields and Records report 148, 631
- Fields Referenced by PeopleCode Programs report 148, 632
- file
  - length 42
- file fields
  - adding to file layouts 508, 510, 528
  - converting text to uppercase 521
  - entering default values 519
  - entering descriptions 519
  - entering the decimal precision 519
  - including/excluding 522
  - interpreting backslashes as escape characters 515
  - naming 510, 519
  - padding 520
  - parsing 514

- selecting data types 520
- selecting date formats/separators 518
- selecting delimiters 517
- selecting inheritance 519
- selecting qualifiers 517, 520
- selecting XML tags 520
- setting lengths 520
- setting properties 517
- spaces, converting tabs to 514
- spaces, trimming 521
- stripping characters 520
- using read/write methods 511
- File Layout Definition Properties dialog box 513
- File Layout Field Properties dialog box 517
- file layouts
  - applying to data interchanges 502
  - creating definitions 507
  - creating definitions (sample) 526
  - exporting data 525
  - field properties, setting 517
  - fields, adding 508, 510, 528
  - fields, including/excluding 522
  - format, CSV *See Also* CSV files
  - format, FIXED *See Also* FIXED files
  - format, XML *See Also* XML files
  - formats, applying 511
  - formats, flat file 502
  - formats, viewing 514
  - import programs, generating/running 525
  - import programs, understanding 521
  - input data, previewing 522
  - input data, troubleshooting 524
  - naming 510
  - producing a sample 525
  - records, adding 508, 509
  - records, root-level 522, 525
  - reordering nodes 510
  - segments *See Also* segments
  - setting properties 512, 527
  - understanding 29, 501
  - using read/write methods 511
- File Layout Segment Properties dialog box 515
- File Layout window 507
- File menu 14, 324
- file records
  - adding to file layouts 508, 509
  - assigning file record IDs 517
  - assigning identification sequence numbers 517
  - entering descriptions for 517
  - naming 510, 516
  - segment terminator 517
  - selecting delimiters 517
  - selecting qualifiers 517
  - selecting XML tags 517
  - setting maximum combined field lengths 517
  - setting properties 515
  - understanding hierarchy 522
  - using read/write methods 511
- file type codes
  - understanding 44
- filtering
  - setting upgrade view options 16
- Find Definition References process
  - changing field definitions 80
  - deleting definitions 35
  - deleting fields 148
  - deleting page controls 334
  - deleting pages 386
  - finding menus that reference components 403
  - running 16, 36
  - viewing referenced definitions 21
- FIXED files
  - importing fields 522
  - input data, previewing 522
  - input data, troubleshooting 524
  - inserting segments/fields 528
  - outputting numbers to 511
  - properties 503
  - trimming field spaces 521
  - understanding 502, 503
- flat files
  - understanding file layouts 501
  - understanding formats 502
- folders
  - content references 440
  - ordering components within 440
  - Project 52
  - tabs for *See Also* tabs
- fonts
  - setting attributes for specific languages 458
  - setting color, style, variant and weight 458
  - setting families 457
  - setting for grids 305
  - setting for page controls 341
  - setting in style sheets 456
  - sizing 457
  - sizing page controls 341
  - understanding generic/installed 455
  - using default style classes 591
  - using style class attributes 455
- footers
  - printing in reports 387
  - setting borders for ChartFields 594
  - setting footer areas for grids 299
  - setting footer areas for scroll areas 275
- Format Families dialog box 86
- Formats dialog box 88
- forms 99
- frames
  - grouping page controls 208
  - inserting 326
  - understanding performance issues 373
  - using 212
  - using default style classes 591
- Free form style sheet
  - editor *See Also* Free form style sheet editor
- Free form style sheet editor
  - understanding 27
- free form sub style sheets
  - changing font and color settings 473
  - understanding 448
- functional indexes
  - DB2 UDB 575
  - Informix 575
  - MSSQL Server 573
  - platform support 177
- functions
  - %EditTable 203
  - Alter *See Also* Alter process
  - ComponentChanged 262
  - Convert Scroll to Grid 279
  - DoModal 211
  - Generate HTML
    - See Also* Generate HTML function

- GetHTMLText 498
- iScripts *See Also* iScripts
- naming records containing 124
- ScrollSelect 303
- SetDBFieldNotUsed 120
- View in Browser
  - See Also* View in Browser function

## G

- Generate HTML function
  - understanding 325
  - using 378
- GetHTMLText function 498
- GIF images
  - converting 493
  - importing 491
  - understanding 488
- Graphics Interchange Format (GIF)
  - See* GIF images
- grid
  - summary properties 300
- Grid Properties dialog box
  - setting body area properties 297
  - setting general properties 290
  - setting header area properties 296
  - setting use properties 301
- grids
  - button images for 619
  - buttons, providing 255
  - collapsing to header bars 307
  - column heading properties, setting 299
  - columns, deleting 284
  - columns, freezing 288
  - columns, inserting/manipulating 283
  - columns, moving 285
  - columns, resizing 285
  - columns properties, setting 285
  - converting from scroll bars 279
  - default settings 281
  - displaying/hiding 326
  - dragging record definitions into 330
  - enabling/disabling personalizations 289
  - HTML tag 378
  - inserting 327
  - inserting/resizing 283
  - inserting page controls 281
  - inserting subpages 238
  - label properties, setting 291
  - labels, aligning column 347
  - nesting in scroll areas 317
  - ordering 369
  - ordering in bulk 628
  - personalizing at runtime 282
  - processing in deferred mode 303
  - resizing in bulk 628
  - rows, controlling styles for 305
  - rows, inserting multiple 303, 304
  - row selection indicators, setting 304
  - selecting pop-up menus 305
  - setting body area properties 297
  - setting display options 305, 306
  - setting embedded pop-up help properties 296
  - setting footer area properties 299
  - setting general properties 290
  - setting header area properties 293

- setting scroll bar options 305
  - setting the View All feature 273, 297
  - setting use properties 301
  - tabs, creating grids with 287
  - tabs, setting styles for 614
  - tab separators, inserting 327
  - understanding 212, 280
  - understanding performance issues 373
  - using default style classes 591
  - using multiple 289
  - using page layout grids 327
  - using the Page Control Inspector 334
- group boxes
  - associating with multicurrency processing 216
  - changing labels 214
  - collapsing 216
  - configuring embedded help 215
  - grouping page controls 208
  - hiding fields 216
  - inserting 213, 326
  - moving 213
  - record properties 214
  - setting properties 214, 216
  - understanding performance issues 373
  - using default style classes 591

## H

- Header Navigation Bar Properties dialog box 293
- headers
  - collapsing grids to header bars 307
  - printing for reports 387
  - setting backgrounds for ChartFields 594
  - setting for scroll areas 269
  - setting header areas for grids 293
- Help menu 19
- history
  - Change Control
    - See Also* Change Control history
  - displaying/updating rows for component records 405
  - effective dates for data rows 101
  - enabling history check boxes on search pages 408
- homepages
  - registering pagelets 440
  - setting templates for homepage pagelets 440
  - using default style classes 591
- horizontal rules
  - drawing 208, 217
  - inserting 326
  - ordering tabs 218
  - setting properties 217
  - using default style classes 591
- hot keys
  - disabling 409
  - invoking spell check 343
  - refreshing pages 200, 421
  - saving pages 406
  - viewing related displays 203
- hover text
  - programmatically changing 263
- HTML
  - converting page controls 378
  - definitions, creating 497

- definitions, naming 498
  - definitions, opening 498
  - definitions, referencing 498
  - definitions, understanding 497
  - editor *See Also* HTML editor
  - HTML areas *See Also* HTML areas
  - using the Generate HTML function
  - See Also* Generate HTML function
  - viewing for pages 379
- HTML Area Properties dialog box 249
- HTML areas
  - changing labels 250
  - generating trees in 246
  - HTML definitions, referencing 498
  - HTML definitions, understanding 497
  - inserting 248, 327
  - populating 247, 249
  - understanding 211, 246
- HTML editor 24
  - understanding 27
- I**
- image catalog 488
  - general images 618
  - lookup page images 620
  - portal images 622
  - query images 620
  - scroll/grid action buttons 619
  - search page images 620
  - tab images 620
  - trees images 621
- image control
  - using controls 618
- image definitions
  - JPG *See Also* JPG images
  - supported file types 488
- image fields
  - creating 65
  - setting attributes 73
  - understanding 65, 73, 617
- image file types
  - BMP 488
  - DIB 488
  - GIF 488
  - JPG 488
  - PNG 488
  - WBMP 488
- ImageReference fields
  - creating 65
  - setting attributes 73
  - understanding 65, 73, 617
- images
  - associating with records 220
  - background images, referencing URLs for 464
  - background images in style sheets, setting 462
  - changing display sizes 492
  - classic tabs, naming images for 611
  - consolidating 494
  - creating alternate types 491
  - definitions, creating 489
  - definitions, opening 491
  - definitions, understanding 487, 617
  - definitions, updating 491
  - drawing image frames 208
  - formats, converting 493
  - formats, selecting 73, 488, 489, 490
  - formats, setting storage 23, 492
  - GIF *See Also* GIF images
  - image fields *See Also* image fields
  - ImageReference fields
    - See Also* ImageReference fields
  - importing 491
  - inserting 327
  - labels, button/link 259
  - labels properties, setting 220
  - mirroring 490
  - naming 489
  - printing a snapshot of your page design 387
  - setting properties 220, 490
  - static *See Also* static images
  - tabs, creating images for 612
  - understanding 487
  - using default style classes 591
  - using in workflow maps 490
  - WBMP *See Also* WBMP images
- image storage 488
- import process *See* import program
- import program
  - generating/running 525
  - understanding 521
- indexes
  - custom, adding 179
  - custom, understanding 176
  - editing DDL 180
  - functional *See Also* functional indexes
  - ordering keys 178
  - search 177
  - setting Alter process options 190
  - specifying creation options 167
  - supported platforms 177
  - understanding 175
  - understanding DDL 156
  - using record DDL 195
  - viewing DDL 197
  - when to create (based on record changes) 177
- Informix
  - altering tables – impact on views 186
  - converting table data 182
  - functional indexes 177, 575
  - maximum image field size 73
  - performing Alter in Place operations 191
  - saving script files 172
- inheritance
  - file fields 519
- Insert into Project dialog box 55
- Insert menu 16, 325, 332, 397
- Install process 3
- instant messaging 254
- interactive mode
  - setting for components 410
  - triggering server processing 200
  - understanding 199
  - using the Expert Entry feature 420
- interfaces
  - component *See Also* component interfaces
  - hiding the Application Designer 13, 161
  - multiple document interface (MDI) 38
- International Organization for Standardization (ISO) 8601 standards 511
- Internet

- Internet scripts *See Also* iScripts
- setting component properties 406
- iScripts
  - naming records 124
  - registering 436
  - understanding 435
- ISO 8601 standards 511

## J

- Japanese characters 68
- javascript
  - in HTML editor 24
- joins 194
- Joint Photographic Experts Group (JPEG)
  - See* JPEG images
- JPG images
  - converting other formats to 493
  - understanding 488
- JSR168STYLEDEF\_SWAN sub-style sheet 477
- JSR168STYLEDEF sub-style sheet 475

## K

- key fields
  - descending keys 136
  - effective dates and level-based controls 318
  - identifying 135, 396
  - ordering 134, 367
  - setting search keys 136
  - understanding 120, 134
  - using in child/parent tables 132
  - using multiple keys in records 132
  - using records without keys 133
- key lists
  - understanding 318
  - using data from current page for searches 256
- keys
  - accessing pages 318
  - alternate search
    - See Also* alternate search keys
  - duplicate order *See Also* duplicate order keys
  - fields *See Also* key fields
  - indexes *See Also* indexes
  - maximum number 177
  - normalized databases 99
  - ordering 134, 178
  - search records *See Also* search records
  - setting access keys 401
  - sharing tables 105
  - using definition keys 556
  - using drop-down list boxes 203
  - using for views 319
  - viewing key-related field characteristics 120

## L

- labels
  - buttons/links, applying image labels to 259
  - buttons/links, applying text labels to 257
  - check boxes, labelling 222

- component tabs, labelling 400
- content references, labelling 440
- drop-down lists, labelling 223
- edit boxes, labelling 225
- field label IDs, setting 138
- fields, deleting labels for 77
- fields, labelling 76
- grids, labelling 291
- group boxes 214
- horizontal rules, labelling 217
- HTML areas, changing for 250
- images, labelling 220
- menus, labelling 425
- page controls, aligning for 347
- page controls, displaying text for 345, 346
- page controls, inserting line feeds in labels for 346
- page controls, labelling 344, 346, 348
- page controls, labelling invisible 350
- page controls, labelling related display 348
- page controls, moving for 347
- page controls, moving labels for 335
- page controls, selecting label IDs for 346
- page controls, using blanks for 345
- pages, positioning on 326
- radio buttons, applying 224
- radio buttons, labelling 223
- record fields, labelling 77
- scroll areas, labelling 269
- scroll bars, labelling 277
- static text, labelling 221
- subpages, labelling 238
- understanding multiple 75
- label translation buffer overlap preventing 336
- languages
  - codes for 586
  - setting font attributes 458
  - setting related language records 128
  - support for spell check 343
  - translations, database 18
- Layout Grid dialog box 327
- Layout menu 325
- lexical characters 83, 90
- library, PeopleSoft Online 19
- line feeds (for page labels) 346
- links
  - activating via the Enter key 262
  - adding prompts 254
  - associating with PeopleCode commands 254
  - destination pages, selecting 256
  - destination pages, selecting action modes for 256
  - destinations, selecting components/menus for 256
  - destination types, setting 252
  - displaying for components 410
  - enabling for display-only access 261
  - external, setting as 252
  - external, setting URLs for 255
  - hover text, setting 263
  - HTML entries for 378
  - indicating link status to users 468
  - inserting 251, 327
  - internal, selecting nodes for 255
  - internal, selecting portals for 255
  - internal, setting as 253
  - labels, setting image 259

- labels, setting text 257
- page anchors, jumping to 254
- page data for searches, using current 256
- secondary pages, accessing 255
- secondary pages, associating primary pages with 245
- understanding 211, 250
- using default style classes 591
- using for toolbar actions 255
- using pseudo classes 468
- list-item markers, setting styles for 467
- literal characters 83, 90
- logging
  - Build process 158
  - sample output for Build logs 169
  - setting build options 167, 168
  - setting levels 168
  - spell check results 45
- long character fields
  - converting table data 182
  - creating 65
  - positioning via builds 163
  - setting attributes 69
  - understanding 65, 69
- long edit boxes 225
  - as rich text fields 225
  - configuring as rich text 356
  - enabling 1000 separators 342
  - enabling spell check 343
  - expanding at runtime 350
  - HTML entries for 378
  - inserting 327
  - labels, applying 225
  - labels, enabling/disabling colons in 348
  - labels, positioning 347
  - replacing blank spaces with fill characters 343
  - sizing 341
  - understanding 210
  - using default base styles 352
  - using default style classes 591
  - using display-only 225, 352
  - using rich text 210
  - wrapping long words 351
- Long Edit Box Properties dialog box
  - See* Edit Box Properties dialog box
- lookup buttons *See* prompt buttons
- lookup pages 620

## M

- Maintain Record DDL dialog box 195
- margins
  - setting for page reports 388
  - setting for record definition reports 150
  - style classes, setting attributes for 455, 465
  - style classes, using default 591
- MDI 38
- menu import process 429
- Menu Listing report 415, 428, 632
- Menu Properties dialog box 426
- menus
  - Application Designer 14
  - Build menu 17, 157
  - component menus, creating custom 424
  - components, adding 427, 437
  - components, finding menus that reference 403
  - components in navigation, including 406
  - Debug menu 17
  - defining transfer 432
  - definitions, copying 427
  - definitions, printing 428
  - definitions, renaming 427
  - Edit menu 15
  - entering descriptions for 426
  - File menu 14
  - Help menu 19
  - importing custom menu groups into portals 429
  - Insert menu 16, 332
  - item properties, setting 425
  - items, deleting 428
  - item types, selecting 425
  - Menu Listing report 632
  - modifying 426
  - naming 425
  - pop-up *See Also* pop-up
  - selecting for link destinations 256
  - setting properties 426
  - setting up security 428
  - Tools menu 17
  - understanding 29, 423
  - using dynamic 40
  - viewing modification history 426
  - View menu 16, 42
  - Window menu 19, 38
- Merge Projects dialog box 57
- message catalog
  - accessing 419
  - setting button/link labels 258, 260
  - setting image labels 220
  - understanding spell check errors 45
  - using static text 209
- messaging
  - instant messaging 254
  - message catalog *See Also* message catalog
  - publishing messages in PeopleCode 202
  - setting error message styles 594
  - suppressing system message boxes 13
  - understanding message channels 29
  - understanding message definitions 29
- meta-SQL 194
- meta characters 83, 90
- methods
  - Alter *See Also* Alter process
  - ReadRowset 507, 511, 512
  - WriteRecord 511, 517, 525
  - WriteRowset 511, 517, 525
- Microsoft Excel *See* MS Excel
- Microsoft SQL Server *See* MSSQL Server
- mobile applications
  - setting system IDs 61
  - using record auto-update options 129
- mobile pages
  - registering 442
  - understanding 29
- modal secondary pages 386
  - guidelines for 241
  - understanding 241
- modes
  - deferred *See Also* deferred mode
  - Edits Display 121
  - Field Display 119



- interactive *See Also* interactive mode
- PeopleCode Debugger
  - See Also* PeopleCode Debugger
- PeopleCode Display 121
- processing modes *See Also* processing modes
- selecting action modes for link destination
  - pages 256
- selecting action modes for search pages 408
- selecting action modes for transfer menus
  - 433
- setting quiet mode 13
- Use Display 120
- View in Browser
  - See Also* View in Browser function
- Modify Order of Page Grid Fields dialog box 628
- MS Excel
  - downloading grid data 296
  - parsing file fields 514
  - parsing support for CSV files 506
- MSSQL Server
  - altering tables – impact on views 186
  - functional indexes 177, 573
  - maximum image field size 73
  - performing Alter in Place operations 191
  - saving script files 172
- multiple document interface (MDI) 38
- My Reports
  - SWAN style sheet 477

## N

- navigation
  - components/pages 394, 401, 410
  - including components in menu navigation
    - 406
  - setting header areas for grids 293
  - using command line parameters 531
  - using default style classes 591
  - using the pagebar 411
- nodes
  - reordering file layout nodes 510
  - selecting for internal links 255
  - setting for the base URI 48
  - Tree Manager buttons 621
- normal forms (databases) 98
- normalized tables 98
- notifications, enabling user 410
- numeric fields
  - creating 65
  - enabling 1000 separators 342
  - inserting decimals automatically 343, 344
  - setting attributes 70, 181
  - setting to zero 342
  - understanding 65, 70

## O

- Open Definition dialog box
  - deleting fields 82
  - opening definitions 30
  - renaming fields 81
- optimization
  - selecting delete records 128
  - understanding problem types 30

- Options dialog box
  - configuring browser options 48
  - setting PeopleTools options 22
  - setting project options 59
- Oracle
  - altering tables – impact on views 186
  - converting table data 182
  - maximum character field size 70
  - maximum image field size 73
  - performing Alter in Place operations 191
  - running alter scripts on tables with LONGS
    - 184
  - saving script files 172
  - using PL/SQLF 181, 184
- output window
  - activating/deactivating 16
  - navigating to build errors 171
  - understanding 14, 21
  - viewing build log output 168
- owner IDs
  - setting for components 403
  - setting for content references 440
  - setting for definitions 31
  - setting for fields 79
  - setting for menus 426
  - setting for pages 382
  - setting for projects 55
  - setting for records 127
  - setting the definitions default 23

## P

- PAALTERNATE\_SWAN sub-style sheet 477
- PAALTERNATE sub-style sheet 475
- pagebars, defining 411
- Page Control Inspector 333
- page control labels
  - aligning 325, 326
  - dashed lines around 336
  - translation buffers 336
  - translations 336
- page controls
  - accessing PeopleCode 373
  - adding static text 209
  - aligning 325, 326, 341, 371
  - applying anchor tags 355
  - associating with multicurrency processing
    - 350
  - auto size fields 326
  - buttons *See Also* buttons
  - charts *See Also* charts
  - check boxes *See Also* check boxes
  - converting to HTML 378
  - copying/moving within/between pages 335
  - deferred mode, enabling 386
  - deleting 334
  - display control and related fields, positioning
    - 363
  - display control and related fields, using 362
  - display controls, ordering 369
  - displaying currency symbols 342
  - drop-down list boxes
    - See Also* drop-down list boxes
  - edit boxes *See Also* edit boxes
  - enable when page is display only 351
  - enabling spell check 343

- finding in page definitions 370
- frames *See Also* frames
- function and data processing 211
- grids *See Also* grids
- group boxes *See Also* group boxes
- grouping 208
- hiding password entries 342
- horizontal rules *See Also* horizontal rules
- HTML areas *See Also* HTML areas
- images *See Also* images
- inserting 326
- inserting decimals automatically 343, 344
- invisible, setting as 349
- invisible controls, creating 364
- invisible controls, modifying 350
- invisible controls, setting labels for 350
- labels, applying 344
- labels, applying related display 348
- labels, moving 335
- labels, setting invisible control 350
- level-based, understanding 314
- level-based, understanding nested 315
- level-based controls and effective dates 318
- links *See Also* links
- long edit boxes *See Also* long edit boxes
- manipulating 332
- naming (system only) 355
- order, using the default 370
- ordering 325, 365
- ordering display controls 369
- ordering logically 367
- ordering via the order tab 369
- ordering visually 366
- order rules, applying 368
- order testing 368
- overlapping 370
- persist grid column in menu 352
- populating automatically 342
- printing a list of 387
- properties, setting general 354
- properties, setting use 348
- push buttons and links 250
- radio buttons *See Also* radio buttons
- related control field 352
- scroll areas *See Also* scroll areas
- scroll bars *See Also* scroll bars
- secondary pages *See Also* secondary pages
- selecting 332
- setting control relationships between fields 350
- setting record/field properties 337
- setting styles 451
- setting to display-only 352
- setting to display only 349
- sizing 333, 341
- spacing 365, 370
- static images *See Also* static images
- static text *See Also* text
- style classes, setting 341
- style classes, using 449
- style classes, using default 591
- subpages *See Also* subpages
- trees *See Also* trees
- understanding 207
- understanding performance issues 372
- using aesthetic 208
- using data-entry 209
- using the Page Control Inspector 333
- using the page control toolbar 326, 331
- wrapping long words 351
- page definition report 388
- pagelets
  - moving/removing 622
  - registering homepage 440
  - setting templates for homepage pagelets 440
  - sizing pages 385
- Page Listing report 632
- Page Properties dialog box
  - changing general information 382
  - changing use information 383
  - defining secondary pages 242
- pages
  - accessing PeopleCode 373
  - accessing via keys 318
  - autosizing 325
  - browsers, verifying page design in 375
  - browsers, viewing in 375
  - buttons *See Also* buttons
  - charts *See Also* charts
  - check boxes *See Also* check boxes
  - components *See Also* components
  - creating 329
  - deferred mode, designing transactions for 201
  - deferred mode, enabling 386
  - definition reports, reading 388
  - definitions, cloning 328
  - definitions, creating 11
  - definitions, identifying online 379
  - definitions, printing 387
  - definitions, understanding 313
  - deleting 386
  - designing inquiry pages 371
  - displaying multiple data occurrences 319
  - dragging record definitions onto pages 330
  - drop-down list boxes
    - See Also* drop-down list boxes
  - edit boxes *See Also* edit boxes
  - field attributes, bulk modifications of 629
  - field definitions, dragging 330
  - fields, bulk insertion of 627
  - fields, bulk modification of 628
  - fields, deleting 148
  - fields, dragging 330
  - fields (record), bulk insertion of 626
  - frames *See Also* frames
  - grids *See Also* grids
  - group boxes *See Also* group boxes
  - homepages *See Also* homepages
  - HTML, generating 378
  - HTML, viewing 379
  - HTML areas *See Also* HTML areas
  - images *See Also* images
  - levels and runtime processing 318
  - links *See Also* links
  - long edit boxes *See Also* long edit boxes
  - lookup pages *See Also* search pages
  - maximizing performance 372
  - mobile *See Also* mobile pages
  - navigating in components 394
  - optimizing performance 199
  - page controls *See Also* page controls
  - page definition toolbar 325
  - personalizing end user pages 323
  - producing 381
  - properties, setting general 382

- properties, setting use 383
- radio buttons *See Also* radio buttons
- refreshing 200
- renaming 386
- report, Fields and Pages 631
- report, Page Listing 632
- report, Pages with PeopleCode 632
- report, Records and Pages 633
- reports, reading definition 388
- saving 386
- scroll areas *See Also* scroll areas
- scroll bars *See Also* scroll bars
- search *See Also* search pages
- secondary *See Also* secondary pages
- setting styles 385
- setting style sheets 479
- sizing 384
- sizing for hidden fields 386
- subpages *See Also* subpages
- understanding 29
- understanding layout 334
- understanding spell check errors 47
- using default style classes 591
- using derived/work fields 322
- using hidden 323
- using prompts 320
- using related displays 203
- using sensitive data 323
- using static text 221
- Pages with PeopleCode report 632
- parent tables 132
- passwords
  - command line parameters 13, 162
  - hiding user entries 342
- PeopleCode
  - Debugger *See Also* PeopleCode Debugger
  - definitions, accessing PeopleCode in page 373
  - definitions, viewing for 16
  - definitions, viewing related 24
  - definition types/IDs 560
  - deleting records 149
  - Display mode 121
  - editing in subrecords 126
  - editor *See Also* PeopleCode editor
  - opening PeopleCode for components 396
  - publishing messages 202
  - report, Fields Referenced by PeopleCode Programs 632
  - report, Pages with PeopleCode 632
  - report, PeopleCode Programs and Field References 632
  - searching for text strings 16
  - selecting PeopleCode menu items 425
  - understanding 24
  - using deferred mode 201
  - validating 18
  - viewing program types 121
- PeopleCode Debugger
  - running with undocked windows 42
  - starting 17
  - viewing errors 21
- PeopleCode Display mode 121
- PeopleCode editor
  - understanding 25
- PeopleCode Editor
  - viewing program types 121
- PeopleCode events *See* events
- PeopleCode functions *See* functions
- PeopleCode Programs and Field References report 632
- PeopleSoft Application Designer
  - See* Application Designer
- PeopleSoft Application Engine programs
  - See* Application Engine programs
- PeopleSoft Data Mover *See* Data Mover
- PeopleSoft EDI Manager *See* EDI Manager
- PeopleSoft financial applications 625
- PeopleSoft Online Library 19
- PeopleSoft Process Scheduler
  - See* Process Scheduler
- PeopleSoft Pure Internet Architecture
  - converting page controls to HTML 378
  - supporting button/link features 250
  - supporting style class attributes 455
  - understanding applications 9
- PeopleSoft splash screen 13
- PeopleSoft Tree Manager 621
- PeopleTools
  - administering DDL 195
  - definition types/IDs 557
  - PeopleCode *See Also* PeopleCode
  - setting options 22
  - tables *See Also* PeopleTools tables
  - using the default style sheet 475, 477
  - viewing release, database, server and user information 19
- PeopleTools tables
  - synchronizing with system catalog tables 164
  - understanding/viewing 97
- performance issues
  - designing pages and controls 372
  - optimizing page performance 199
  - ordering index keys 178
  - using prompts/drop-down lists for grids 281
- permission lists
  - adding components 438
  - understanding 435
- persistent search configuration
  - subrecord considerations 127
- Persist Grid Column in Menu
  - custom application search pages 262
- persist in menu
  - subrecord considerations 127
- personalizations, grid 289
- Personalize Column and Sort Order page 282
- PL/SQL 181, 184
- pop-up menus
  - associating with page fields 431
  - button images for 429, 619
  - creating 430
  - defining transfer menu items 432
  - selecting for grids 305
  - understanding 429
  - using 40
- pop-up pages *See* pop-up pages
  - configuring 263
  - creating 229
  - creating context sensitive help 231
  - identifying 225
  - inserting controls for 230
  - sizing 230
  - understanding 226
  - using edit boxes as 353
  - using edit boxes to control 230
  - using links to access 263

- using message catalog with 231
  - using push buttons to access 263
- popup pages *See* pop-up pages
- Portable Network Graphic (PNG)
  - See* PNG images
- Portal Registry
  - adding components 437
  - spell check errors 46
  - understanding 435
- portals
  - assigning components 436
  - button images for 622
  - importing custom menu groups 429
  - Portal Registry *See Also* Portal Registry
  - registering components in multiple 442
  - selecting for internal links 255
  - using default style classes 591
  - viewing pages in 384
- PPLTOOLS project 98
- printing
  - button image for 619
  - list of page controls and parameters 387
  - menu definitions 428
  - page definitions 387
  - page design snapshot 387
  - page report headers, footers and borders 387
  - projects 15
  - record definitions 149
  - setting page margins 388
- Print Setup dialog box 387
- procedural language/Structured Query Language (PL/SQL) 181, 184
- processes
  - Alter *See Also* Alter process
  - batch *See Also* Application Engine programs
  - Build *See Also* Build process
  - business 28
  - Create Table 163
  - Find Definition References
    - See Also* Find Definition References
  - process
  - import *See Also* import program
  - Install 3
  - menu import 429
  - processing groups 406
  - processing modes 199
  - Process Scheduler
    - See Also* Process Scheduler
  - Upgrade Compare
    - See Also* Upgrade Compare process
  - Upgrade Copy
    - See Also* Upgrade Copy process
- processing groups 406
- processing modes
  - setting based on users 420
  - setting for components 410
  - understanding 199
- Process Scheduler
  - running Crystal reports 633
  - running processes 254
- Process Scheduler Request page 633
- production database 565
- profiles
  - access *See Also* access profiles
  - user 433
- programming
  - languages 24
- programs
  - Application Engine
    - See Also* Application Engine programs
  - Fields Referenced by PeopleCode Programs
    - report 148, 632
  - PeopleCode Programs and Field References
    - report 632
- Project Properties dialog box 54
- projects
  - building 534
  - building records 17
  - building record subsets 159
  - building via the Build menu 159
  - building via the command line 161
  - change *See Also* change projects
  - comparing via command line parameters 542
  - creating 54
  - definitions, accessing 52
  - definitions, automatically locking 567
  - definitions, inserting 17, 55, 60
  - definitions, inserting comments into 568
  - definitions, locking 566, 567
  - definitions, locking all 567
  - definitions, modifying 54
  - definitions, removing 57
  - handling errors for builds 161
  - locking 552
  - locking all definitions 567
  - locking definitions 566, 567
  - locking definitions automatically 567
  - merging 15, 57
  - PPLTOOLS 98
  - printing 15
  - project workspace
    - See Also* project workspace
  - properties, editing 15
  - properties, setting 54
  - reloading upon startup 60
  - saving 15, 54
  - setting options 22, 59
  - setting system IDs 61
  - understanding 20, 29, 51, 564
  - Upgrade Compare process
    - See Also* Upgrade Compare process
  - Upgrade Copy process
    - See Also* Upgrade Copy process
  - upgrading application projects 18
  - using Change Control 564
  - validating 17, 21, 22, 60
  - viewing 52
  - viewing messages 21
- project workspace
  - activating/deactivating 16
  - displaying/hiding 325
  - understanding 20, 52
  - Upgrade view *See Also* Upgrade view
  - viewing definitions 31
- prompt buttons
  - adding 254
  - enabling 342
  - understanding 320
- prompt dialogs
  - allowing Search events 137
  - enabling PeopleCode Search events 137
- prompt tables
  - button image for 619
  - defining records as 146
  - naming records as 123
  - setting for fields 121, 145

- translate tables *See Also* translate tables
- using deferred mode 203
- using effective dates 102
- PSACE\_SWAN sub-style sheet 477
- PSACE sub-style sheet 475
- PSALTERNATE\_SWAN sub-style sheet 477
- PSALTERNATE sub-style sheet 475
- PSAUDIT table 130
- PSCHGCTLHIST table 556
- PSCHGCTLLOCK table 556
- PSIDE.EXE *See* command line
- PSSTYLEDEF\_SWAN style sheet 477
- PSSTYLEDEF style sheet 475
- PTSTYLEDEF\_SWAN sub-style sheet 477
- PTSTYLEDEF sub-style sheet 475
- push button/hyperlink page control
  - setting 260
- Push Button/Hyperlink Properties dialog box
  - change hover text 263
  - labelling buttons/links 256
  - setting destination types 252
- push button/hyperlinks
  - setting mouse over pop-up properties 263
- push buttons
  - modifying SWAN 485

## Q

- queries
  - button images for 620
  - specifying query security records 128
  - views *See Also* query views
- query views
  - defining records as 118
  - naming records as 124
- quiet mode 13

## R

- radio buttons
  - enable when page is display only 351
  - HTML entries for 378
  - inserting 327
  - linking with records 224
  - ordering 369
  - setting label properties 224
  - setting use properties 224
  - understanding 210, 223
  - using default style classes 591
  - using in deferred mode 203
- ReadRowset method 507, 511, 512
- reasonable dates 146
- recent search results
  - search records 415
- recent search results dialog box 141
- recent search results page construction
  - understanding 418
- recent search results process 417
- record definition report 150
- Record Field Properties dialog box
  - selecting field labels 77
  - setting field edit properties 143
  - setting field use properties 134
  - viewing editing options 121
- viewing key-related characteristics and
  - default values 120
- record fields
  - including in persistent search results 140
- record field table (RFT) *See* RFT
- record groups
  - defining 111
  - understanding 106
- record properties
  - group boxes 214
- Record Properties dialog box 127
- records
  - adding descriptions 127
  - adding SetIDs 109
  - altering 17
  - Alter process *See Also* Alter process
  - associating with check boxes 222
  - associating with images 220
  - audit options, setting 137
  - audit records, creating 130
  - audit records, setting user-defined 128
  - audits, using record-level 128
  - building 17
  - building subsets 159
  - creating 124
  - definition, building the active 159
  - definitions, creating 10
  - definitions, deleting 148
  - definitions, dragging 330
  - definitions, planning 100
  - definitions, printing 149
  - definitions, renaming 148
  - displaying/updating rows for component
    - records 405
  - dynamic views *See Also* dynamic views
  - effective dates, understanding 100
  - effective dates, using 405
  - field (record) edit properties, setting 143
  - field definitions, dragging 330
  - field definitions, viewing basic 119
  - field label IDs, setting 138
  - field labels, selecting 77
  - field properties, understanding 131
  - fields, bulk insertion of 626, 627
  - fields, bulk modifications of 628
  - fields, defining set control 109
  - fields, deleting 147
  - fields, dragging 330
  - fields, identifying key 135
  - fields, inserting 125
  - fields, moving 146, 147
  - fields, reordering 122
  - fields, setting autocomplete 139
  - fields, setting related date 140
  - fields, setting search persistence for custom
    - search pages 140
  - fields, setting time zone 140
  - fields, understanding 63, 125
  - fields (record), bulk insertion of 626
  - field types, changing 80
  - field use properties, setting 134
  - field values, setting default 139
  - file records *See Also* file records
  - groups *See Also* record groups
  - indexes, creating 176
  - indexes, when to create 177
  - key-related characteristics and default values,
    - viewing 120

- key fields, identifying 135
- keys, ordering 134
- keys, using duplicate order 133
- keys, using multiple 132
- keys, using no 133
- keys on parent/child tables, using 132
- linking with radio buttons 224
- naming 123, 127
- navigating to build errors 171
- opening 125
- properties, setting 127
- properties, setting drop-down list record 222
- properties, setting field use 134
- properties, setting record field edit 143
- properties, understanding field 131
- properties for page controls, setting 337
- query views *See Also* query views
- relationship to scroll areas/bars 266
- report, Fields and Records 148, 631
- report, record definition 150
- report, Records and Fields 631
- report, Records and Pages 633
- saving 123
- search *See Also* search records
- securing access 128
- selecting delete records for optimization 128
- setting parent 128
- setting related language 128
- setting tablespaces 196
- specifying relationship settings 128
- SQL tables *See Also* SQL tables
- SQL views *See Also* SQL views
- subrecords *See Also* subrecords
- synchronizing with application data tables 164
- temporary tables *See Also* temporary tables
- understanding 30, 115
- understanding DDL 156
- understanding performance issues 373
- using auto-update options 129
- using subpages 237
- using third normal form 100
- viewing 119
- viewing editing options 121
- viewing modification history 127
- viewing PeopleCode program types 121
- viewing records associated with set controls 113
- work *See Also* work records
- worklist *See Also* worklist records
- Records and Fields report 631
- Records and Pages report 633
- refreshing
  - pages 200
  - Refresh buttons in components, including 420
  - related displays 203
- Registration Wizard
  - accessing 436
  - registering components 11, 436, 442
  - registering iScripts 436
  - registering mobile pages 442
  - setting menu item properties 425
  - understanding 435
- related content
  - context search record 405
  - key field configuration 405
- related display fields
  - labelling 348
  - ordering display controls 369
  - understanding 203
  - using 362
- related edit fields, creating 363
- release numbers, available 588
- Rename Definition dialog box 34, 81
- reports
  - Applications and Fields 632
  - Change Control information 556
  - creating compare reports via command line parameters 542
  - cross-reference *See Also* Crystal reports
  - Field Listing 632
  - Fields and Pages 631
  - Fields and Records 148, 631
  - Fields Referenced by PeopleCode Programs 148, 632
  - generating ad hoc 633
  - Menu Listing 632
  - page definition 388
  - Page Listing 632
  - Pages with PeopleCode 632
  - PeopleCode Programs and Field References 632
  - PeopleTools Menu Listing 415, 428
  - printing 15
  - record definition 150
  - Records and Fields 631
  - Records and Pages 633
  - reporting Change Control history 562
  - upgrade reports, printing 15
  - viewing 15
  - viewing a print preview 15
  - viewing for definitions 15
  - Window Listing 633
- RFT
  - setting button/link labels 253, 258, 259
  - setting field names 66, 346
  - setting image labels 220
- rich text editor
  - adding plug ins to the configuration file 641
  - configuring 210
  - configuring long edit boxes to display 356
  - creating custom buttons 639
  - creating plug in files 639
  - enabling 225
  - long edit box as 225
  - multilingual support 640
  - using images with plug ins 640
- rich text fields *See Also* long edit boxes
- rows
  - adding multiple rows in deferred mode 204
  - grids, controlling row styles for 305
  - grids, disabling row insertion/deletion for 303
  - grids, inserting multiple rows into 303, 304
  - grids, setting fixed row height for 306
  - grids, setting odd/even row style for 306
  - grids, setting row selection indicators for 304
  - scroll bars, deleting rows from 279
  - scroll bars, disabling row insertion/deletion for 278
  - scroll bars, identifying rows in 279
  - scroll bars, inserting rows into 279
  - scroll bars, navigating in 279

# S

## scripts

- building script files 158
- build options, setting 171
- builds, selecting file name options for 174
- builds, selecting file options for 173
- builds, selecting file overwrite options for 173
- iScripts *See Also* iScripts
- modifying invisible page controls via JavaScript 350
- reviewing/updating 160
- running 160
- saving script files 172
- using comments 172

## scroll area

- summary properties 275

## Scroll Area Properties dialog box 276

- setting header area properties 272

## scroll areas

- button images for 619
- buttons, providing 255
- inserting 327
- label properties, setting 269
- labels, displaying 348
- manipulating fields 267
- nesting grids 317
- ordering 369
- processing in deferred mode 303
- processing pages 266
- properties, setting display 306
- properties, setting general 267
- properties, setting header area 269
- properties, setting label 269
- properties, setting use 276
- record relationships 266
- setting body area properties 273
- setting embedded pop-up help properties 272
- setting footer area properties 275
- understanding 211, 266
- understanding performance issues 373
- using default style classes 591

## Scroll Areas Properties dialog box

- setting body area properties 273

## Scroll Bar Properties dialog box 277

## scroll bars

- button images for 619
- buttons, adding scroll action 278
- buttons, providing 255
- controlling scrollable data 279
- converting to grids 279
- inserting 327
- ordering 369
- processing pages 266
- record relationships 266
- setting attributes 278
- setting field use options 278
- setting label properties 277
- setting use properties 277
- understanding 211, 266
- understanding performance issues 373

## ScrollSelect function 303

## searches

- alternate search keys  
*See Also* alternate search keys
- configuring persistent search list box fields

141

definitions and their references

*See Also* Find Definition References

process

PeopleCode programs 16

projects 21

records *See Also* search records

running for components 405

search pages *See Also* search pages

SQL programs 16

## search indexes 177

## search pages

- adding from/through logic 141
- button images for 620
- disabling advanced search features 137
- enabling history check boxes 408
- enforcing required property and table edits 136
- opening new browser windows 411
- setting component options 408
- using instructional text 409

## search persistence

- custom search pages 140

## search records

- adding from/through search logic 141
- designing inquiry pages 371
- key lists *See Also* key lists
- overriding 419
- selecting search fields 137
- setting for components 405
- understanding 413

## secondary page properties

- modality 386

## secondary pages

- accessing via buttons/links 255
- associating with primary pages 245
- default buttons 243
- defining 242
- Display in Modal Window 386
- inserting 327
- inserting controls 244
- modality 241
- modal options 386
- selecting style sheets 244
- sizing 244, 326
- understanding 211, 239

## security

- Change Control 555, 562
- enabling the Expert Entry feature 420, 421
- granting build authority 156
- modifying invisible page controls via JavaScript 350
- performing bulk operations 625
- setting up for menus 428
- specifying query security records 128
- using sensitive data on pages 323

## segments

- inserting into file layouts 509, 528
- setting properties 515
- using 509

## servers

- optimizing transmissions 199
- triggering processing 200
- using deferred mode 200, 410
- using interactive mode 410
- using the Refresh button 420

## set control fields

- assigning 108

- defining 109
  - defining for trees 112
  - enabling 145
  - modifying 110
  - overriding 146
  - understanding 107
  - using associated records 121
  - viewing associated records 113
- SetDBFieldNotUsed function 120
- setIDs
  - adding to records 109
  - assigning default 110
  - creating 111
- Set System Default Colors dialog box 43
- sizing sets 156
- sliding door tabs 481
- SmartNavigation pages
  - customizing 647
- smart punctuation 90
- spell check
  - button image for 619
  - enabling for Application Designer 22, 48
  - enabling for component fields 409
  - enabling for components 406
  - enabling for edit boxes 343
  - launching portal pages 48
  - understanding 44
  - understanding results 45
- splash screen, PeopleSoft 13
- splitter bars 41
- SQL
  - constructing definitions 26
  - DDL *See Also* DDL
  - editor *See Also* SQL editor
  - running as part of the Build process 160
  - searching for text strings in programs 16
  - tables *See Also* SQL tables
  - understanding definitions 30
  - views *See Also* SQL views
- SQL Alter
  - creating indexes 176
  - renaming database table fields 82, 148
  - synchronizing database/PeopleSoft definitions 50
- SQL editor
  - understanding 26
- SQL Editor
  - opening 152
- SQL tables
  - building 11
  - changing field definitions 80
  - creating 50
  - naming 152
  - records *See Also* records
  - understanding the Build process 155
- SQL views
  - creating select statements 152
  - defining records as 118
  - naming records as 123
  - updating select statements 143
- static images
  - adjusting size/shape 219
  - changing dynamically at runtime 73
  - drawing frames 209
  - HTML tag 378
  - inserting 327
  - setting properties 219
  - using controls 618
  - using default style classes 591
- status
  - activating/deactivating the status bar 16
  - indicating link status to users 468
  - viewing in the output window
    - See Also* output window
- style classes
  - accessing attributes 456
  - adding to style sheets 453
  - understanding 449
  - understanding attributes 454
  - understanding pseudo classes 468
  - using custom 451
  - using defaults 450, 591
  - visual effects 470
  - visual formatting 469
- styles 44
  - classes *See Also* style classes
  - default base styles for page controls 352
  - setting for page control labels 346
  - setting for page controls 341
  - style sheets *See Also* style sheets
  - understanding 44
- style sheets
  - background attributes, setting 461
  - background image URLs, referencing 464
  - classes 445, *See Also* style classes
  - colors, changing classic tab 611
  - colors, changing tab 481
  - converting standard to free form 471
  - creating 451
  - creating free form 471
  - creating tab images 612
  - default, changing 475
  - default, using 475, 477
  - fixed form 447
  - font attributes for specific languages, setting 458
  - fonts, setting 456
  - fonts, understanding 455
  - free form sub style sheets 448
  - inserting style classes 451
  - override style classes 451
  - pages, selecting for 385
  - pages, selecting style sheets for secondary 244
  - pages, setting for 479
  - setting border attributes 464
  - setting margins 465
  - setting spacing and alignment 458
  - specifying miscellaneous attributes 467
  - standard 447
  - sub style sheets *See Also* sub style sheets
  - SWAN push buttons 485
  - understanding 30, 445
- sub-style sheets
  - inserting into style sheets 453
  - using default style classes 475, 477, 591
- subpages
  - defining 237
  - inserting 327
  - inserting into grids 238
  - inserting into pages 238
  - setting informational labels 238
  - sizing 326
  - understanding 210, 237
- subrecords
  - defining records as 118



- editing PeopleCode 126
  - inserting into records 126
  - naming 124
  - nesting 126
  - persist in menu property, considerations for
    - using 127
  - printing 152
  - viewing 126
- sub style sheets
  - creating 451
  - understanding 447
- Sybase
  - altering tables – impact on views 186
  - maximum image field size 73
  - performing Alter in Place operations 191
  - saving script files 172
- synchronization
  - Build process *See Also* Build process
  - synchronizing database/PeopleSoft definitions 50
  - synchronizing records with application data tables 164
  - using views online 193
  - when to run the Alter process 187
- system catalog tables
  - creating/understanding 97
  - synchronizing with PeopleTools tables 164
  - understanding the Build process 163
- system IDs, setting 61
- system variables
  - %Market 402
  - entering default field values 139

## T

- tables
  - altering *See Also* Alter process
  - application data
    - See Also* application data tables
  - confirming builds 175
  - control *See Also* control tables
  - creating 164
  - creating audit tables 130
  - defining record groups 111
  - fields with edit tables 202
  - maximum columns 164
  - naming non-standard SQL tables 118
  - PeopleTools 97
  - PSAUDIT 130
  - PSCHGCTLHIST 556
  - PSCHGCTLLOCK 556
  - set control fields, assigning 108
  - set control fields, using 121
  - SetIDs, adding 109
  - setIDs, creating 111
  - setting create options 165
  - sharing 106
  - SQL *See Also* SQL tables
  - system catalog 97
  - tablesets *See Also* tablesets
  - Table Sharing feature 108, 121
  - tablespaces *See Also* tablespaces
  - temporary *See Also* temporary tables
  - translate *See Also* translate tables
  - understanding effective dates 100
  - understanding normalized 98
  - understanding performance issues 373
  - understanding the Build process 163
  - URL Maintenance 255, 464, 490
  - using child/parent 132
  - using key fields 132
  - viewing DDL 197
- tablesets
  - creating TableSet IDs 111
  - defining controls 112
  - defining record groups 111
  - PeopleSoft applications 106
  - sharing record groups 106
  - sharing tables 107
  - sharing trees 112
  - understanding 103
- Table Sharing feature 108, 121
- tablespaces
  - setting 196
  - viewing DDL 197
- tabs
  - displaying for components 410
  - grids, creating tabbed 287
  - grid tab separators, inserting 327
  - images, creating 612
  - images, naming 611
  - navigation buttons 620
  - ordering 218, 367
  - order testing 326
  - setting styles 614
  - specifying component labels 400
  - understanding classic definitions 611
  - understanding definitions 481
  - using default style classes 591
- templates
  - page 329
  - record DDL 195
  - setting for homepage pagelets 440
- temporary tables
  - creating via the Alter process 191
  - defining records as 118
  - naming 123
  - running Alter scripts 184
- test database 565
- Test Format dialog box 91
- testing
  - applications 11
  - field formats 91
  - page control order 368
  - reasonable dates 146
  - tab order 326
  - using the test database 565
- text
  - aligning 221
  - instructional text for search pages, using 409
  - labels, setting button/link text 257, 258, 260
  - labels for page controls, setting text 346
  - label text properties, setting 221
  - multicurrency processing 221
  - setting styles 221
  - static, inserting 327
  - static text, HTML tag for 378
  - static text for page controls, adding 209
- three-tier execution 406
- time fields
  - creating 65
  - displaying time zones 343
  - field length in file layouts 511
  - setting attributes 72

- understanding 65, 72
- time zones
  - adjusting time fields 72, 101
  - displaying for fields 343
  - understanding time zone fields 140
- toolbars
  - activating/deactivating 16
  - Application Designer window 20
  - buttons, adding 255
  - defining for components 409
  - page control 326, 331
  - page definition 325
  - using dockable 41
  - using dynamic 40
- Tools menu 17, 325
- transactions
  - button image for 619
  - components *See Also* components
  - running in deferred mode 201, 202
  - using related displays 203
  - using the Expert Entry feature 420
- transfer menus, defining 432
- translatability options 22
- translate tables
  - adding values 94
  - attributes 93
  - changing values 96
  - deleting values 96
  - effective dates 94
  - enabling for fields 145
  - related display fields 351
  - saving 96
  - understanding 92
  - using 92
- translation
  - base and operator language databases 18
  - entering page control labels 345
- Tree Manager 621
- trees
  - dynamic tree controls (deprecated) 211
  - generating in HTML areas 246
  - inserting 327
  - sharing 112
  - Tree Manager buttons 621
- triggers
  - creating/understanding 192
  - triggering server processing 200

## U

- Universal Resource Identifier (URI), setting the
  - base 23, 440
- Upgrade Compare process
  - locking and upgrades 552
- Upgrade Copy process
  - databases, locking 552
  - databases for development, using multiple 565
- Upgrade Project report 15
- Upgrade view
  - setting filtering 16
  - understanding 52
  - using 53
  - viewing upgrade-only definitions 57
- upgrading applications
  - Change Control history
    - See Also* Change Control history
  - Change Control stamping
    - See Also* Change Control stamping
  - copying projects via command line
    - parameters 536
  - locking *See Also* Change Control locking
  - recommendations 323
  - setting Alter process options 190
  - Tools menu Upgrade item 18
  - Upgrade Copy process
    - See Also* Upgrade Copy process
  - Upgrade view *See Also* Upgrade view
  - viewing actions 21
- URI, setting the base 23, 440
- URL Maintenance table 255, 490
- URLs
  - content references
    - See Also* content references
  - controlling via style classes 455
  - copying to the clipboard 411
  - referencing background image URLs 464
  - setting for images 490
  - URL Maintenance table 464
- Use Display mode 120
- user IDs
  - Change Control implementation
    - considerations 555
  - locked definitions, identifying 551
  - locking, understanding 551
  - sharing IDs 563
  - upgrades and locking 552
  - upgrades and stamping 555
- user profiles 433

## V

- validation
  - components 21
  - fields with edit tables 202
  - PeopleCode 17
  - projects 18, 21, 22, 60
  - WSRP 18
- version control 552
- View in Browser function
  - activating 326
  - testing page control order 368
  - understanding 325
  - using 375
- View in Browser mode
  - See Also* View in Browser function
- View menu 16, 42, 324
- views
  - altering tables 186
  - creating 166, 193
  - Development *See Also* Development view
  - dynamic *See Also* dynamic views
  - query *See Also* query views
  - SQL *See Also* SQL views
  - understanding 192
  - Upgrade *See Also* Upgrade view
  - using joins 194
  - using keys 319

## W

### warnings

- adding to Build logs 168
- displaying in deferred mode 205
- enabling save warnings for grid changes 305
- enabling save warnings for leaving pages 262
- handling for pages/page-controls 386
- image file size 489

### WBMP images

- converting 493
- importing 491
- understanding 488

### Window Listing report 633

### Window menu 19, 38

### windows

- accessing context-sensitive help 19
- activating 38
- Application Designer window components 14
- opening links in new windows 261
- opening links that replace the current window 261
- using dockable 41
- using splitter bars 41
- Window Listing report 633
- Window menu 19

### Windows

- building projects via the command line 159, 161
- functional indexes 177
- saving script files 172
- setting the three-tier execution location 406
- sizing pages 385

### wireless bitmap (WBMP) *See* WBMP images

### workflow maps

- understanding definitions 28
- using images 490

### worklist records

- accessing/viewing 410
- naming 124

### work records

- adding pages based on work records to components 415
- defining records as 118
- naming 123
- naming for SQR reports 124
- referencing for pages 372
- using derived/work fields 322
- using hidden pages 323
- varying prompt tables 146

### workspaces

- definition workspace *See Also* definition workspace
- project workspace *See Also* project workspace
- using splitter bars 41

### WriteRecord method 511, 517, 525

### WriteRowset method 511, 517, 525

### generating 525

### importing fields 522

### properties 506

### selecting XML tags for fields 520

### understanding 502, 506

## X

### XML files

- file layouts, associating tags with 514
- file layouts, displaying the buffer size for 514

