

PeopleSoft®

Enterprise PeopleTools 8.45
PeopleBook: Global Technology

June 2004

Enterprise PeopleTools 8.45 PeopleBook: Global Technology
SKU PT845GBL-B 0604
Copyright © 1988-2004 PeopleSoft, Inc. All rights reserved.

All material contained in this documentation is proprietary and confidential to PeopleSoft, Inc. ("PeopleSoft"), protected by copyright laws and subject to the nondisclosure provisions of the applicable PeopleSoft agreement. No part of this documentation may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, including, but not limited to, electronic, graphic, mechanical, photocopying, recording, or otherwise without the prior written permission of PeopleSoft.

This documentation is subject to change without notice, and PeopleSoft does not warrant that the material contained in this documentation is free of errors. Any errors found in this document should be reported to PeopleSoft in writing.

The copyrighted software that accompanies this document is licensed for use only in strict accordance with the applicable license agreement which should be read carefully as it governs the terms of use of the software and this document, including the disclosure thereof.

PeopleSoft, PeopleTools, PS/nVision, PeopleCode, PeopleBooks, PeopleTalk, and Vantive are registered trademarks, and Pure Internet Architecture, Intelligent Context Manager, and The Real-Time Enterprise are trademarks of PeopleSoft, Inc. All other company and product names may be trademarks of their respective owners. The information contained herein is subject to change without notice.

Open Source Disclosure

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

Apache Software Foundation

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright (c) 1999-2000 The Apache Software Foundation. All rights reserved.

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

OpenSSL

Copyright (c) 1998-2003 The OpenSSL Project. All rights reserved.

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

SSLey

Copyright (c) 1995-1998 Eric Young. All rights reserved.

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

Loki Library

Copyright (c) 2001 by Andrei Alexandrescu. This code accompanies the book:

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

Contents

General Preface

- About This PeopleBookxiii**
- PeopleSoft Application Prerequisites.....xiii
- PeopleSoft Application Fundamentals.....xiii
- Related Documentation.....xiv
 - Obtaining Documentation Updates.....xiv
 - Ordering Printed Documentation.....xiv
- Typographical Conventions and Visual Cues.....xv
 - Typographical Conventions.....xv
 - Visual Cues.....xvi
 - Country, Region, and Industry Identifiers.....xvi
 - Currency Codes.....xvii
- Comments and Suggestions.....xvii
- Common Elements in These PeopleBooksxvii

Preface

- Global Technology Preface.....xix**
- PeopleTools Global Technology.....xix

Part 1 Getting Started

Chapter 1

- Getting Started with Global Technology.....3**
- PeopleTools Global Technology Overview.....3
- PeopleTools Global Technology Implementation.....4

Part 2 Configuring Your System for Global Implementation

Chapter 2

Controlling International Preferences.....11

Understanding International Preferences.....11

Setting Up Language Preferences in the PeopleSoft Pure Internet Architecture.....12

 Understanding the Session Language.....12

 Choosing a Session Language.....12

 Changing the Session Language While Signed In.....13

 Changing the Session Language Programmatically.....13

 Determining the Session Language When the Signon Page is Bypassed.....14

 Controlling the User Profile Language Preference.....14

 Determining the Language of Help Documentation.....15

Applying Browser Language Preferences.....15

 Understanding Browser Language Preferences.....15

 Setting Browser Language Preferences.....15

 Using Browser Language Settings to Determine the Sign-in Language.....15

Setting Up Language Preferences for Microsoft Windows-Based PeopleTools.....17

 Understanding Language Preferences for Microsoft Windows-Based PeopleTools.....17

 Controlling PeopleSoft Configuration Manager Language Settings.....17

Setting Up PeopleSoft Pure Internet Architecture Locale-Based Formatting.....18

 Understanding Locale-Based Formatting Drivers.....19

 Pages Used to Set Up PeopleSoft Pure Internet Architecture Locale-Based Formatting.....20

 Defining User-Specific Formatting.....20

 Defining Locales for Locale-Specific Formatting.....21

 Assigning Locale-Specific Formatting.....22

 Applying System-Wide Default Formatting.....23

Setting up Locale-Based Formatting for Microsoft Windows-based PeopleTools.....23

Chapter 3

Setting and Maintaining Time Zones.....25

Understanding Time Zones.....25

Setting the Base Time Zone.....26

Maintaining Time Zones.....28

 Understanding Time Zone Maintenance.....28

 Viewing Time Zone Definitions.....28

 Generating Time Zone Offsets.....29

 Viewing Daylight Savings Start and End Dates.....30

Controlling Time Zone Display When Developing PeopleSoft Application Objects.....31

 Understanding Time Zone Display Settings.....31

 Setting Time Zone Options in Record Field Properties.....32

Setting Time Zone Options in Page Field Properties.....33
 Using PeopleCode Time Functions.....33

Chapter 4

Selecting and Configuring Character Sets and Language Input and Output.....35
 Understanding Character Sets.....35
 Character Sets.....35
 Common Character Sets.....36
 The Unicode Standard.....38
 Non-Unicode Character Sets.....42
 Character Sets in the PeopleSoft Pure Internet Architecture.....43
 Selecting Character Sets.....50
 Selecting Database Character Sets.....50
 Selecting Application Server Character Sets.....51
 Selecting and Managing Client Workstation Character Sets.....51
 Selecting Email Character Sets.....54
 Validating Field Length.....55
 Understanding Application Designer Field Length Semantics.....55
 Understanding Field Length Checking for Non-Unicode Databases.....57
 Enabling or Disabling Data Field Length Checking.....57
 Handling International Characters in PeopleTools.....58
 Standard Name Formats for Chinese, Japanese, and Korean Characters.....58
 Detecting and Converting Characters.....59

Chapter 5

Controlling Currency Display Format.....61
 Understanding Currency-Specific Settings.....61
 Setting Up Currency Amount Fields When Developing Applications.....62
 Setting Currency Field Display Properties When Developing Applications.....62
 Using System-Wide Multicurrency Settings.....63
 Understanding Multicurrency Settings.....63
 Connecting a Currency Control Field to the Multi-Currency Check Box.....64
 Enabling or Disabling System-Wide Currency Settings.....64
 Resizing Currency Fields by Using the International Field Size Utility.....65

Chapter 6

Running COBOL in a Unicode Environment.....67

Understanding COBOL in a Unicode Environment67

 Unicode Encodings in PeopleSoft COBOL.....67

 Expanded Storage Space Requirements.....68

 Special Logic for Parsing Unicode Strings.....69

 COBOL Sorting.....69

 Unicode-Specific Error Messages.....69

Running the COBOL Conversion Utility.....70

 Understanding the COBOL Conversion Utility.....70

 Running the Conversion Utility.....71

 Identifying Converted COBOL Programs.....72

 Understanding What Is Expanded.....72

 Using Utility Directives.....77

 Viewing Error Logs.....79

Fine-Tuning COBOL Programs.....80

 Identifying Unicode and Non-Unicode Data.....81

 Specifying Column Lengths in Dynamic SQL.....81

 Defining Single Character Arrays.....83

Chapter 7

Sorting in PeopleTools.....93

Understanding Sort Orders.....93

 Sorting Overview.....93

 PeopleTools Sorts.....94

 Database-Level SQL ORDER BY Sorts.....95

 PeopleTools In-Memory Sorts.....96

 Binary Sorts.....96

Setting the Sort Order.....97

Forcing a Binary Sort in SQL.....97

Sorting in COBOL.....98

Sorting in SQR.....99

Chapter 8

Understanding Global Reporting and Data Analysis Tools.....101

Understanding Language-Sensitive Queries.....101

 Scheduled Queries.....102

Understanding PeopleSoft Cube Manager for Global Implementations.....102

Cognos PowerPlay Local-Language Cubes.....102

Hyperion Essbase Multiple-Language Cubes.....103

Understanding the Strings Table and Language-Sensitive Text in Reports.....104

 Evaluation of PS/nVision String Variables.....105

Understanding SQR for PeopleSoft Globalization.....107

 Printing for A4 Paper.....107

 Currency Precision.....108

 Date and Time Formatting.....108

 Report Translation.....109

 The PSSQR.INI and PSSQR.UNX Files.....112

 SQR Configuration for Processing International Text.....112

 SQR Configuration for Printing International Text.....113

 International Text in SQR for PeopleSoft Programs.....122

 SQR for PeopleSoft-Supported Character Set Encodings.....124

Understanding PS/nVision Reporting in a Global Implementation.....137

 PS/nVision Design-Time Globalization Features.....137

 PS/nVision Run-Time Language Features.....139

International Versions of Microsoft Excel.....141

Chapter 9

Developing Global Applications.....143

 Developing Easily Translatable Applications.....143

 Designing Global-Ready Pages.....144

Part 3

Working With PeopleSoft Applications in Multiple Languages

Chapter 10

Using Related Language Tables.....149

 Understanding Related Language Tables.....149

 Related Language Tables Overview.....149

 Related Language Table Structure.....152

 How Related Language Tables Store Translations.....153

 How Related Language Tables Are Used.....154

 Installing PeopleSoft-Provided Translations.....155

 Understanding Translation Installation.....155

 Adding Translations to an Existing Database.....156

Upgrading Your Customizations into a Multilingual Database.....	157
Swapping the Base Language.....	158
Understanding Base Language Swapping.....	158
Prerequisites.....	159
Running the Swap Audit Report.....	159
Checking Space and Resources.....	161
Running the SWAP_BASE_LANGUAGE Data Mover Command.....	161
Creating Related Language Tables.....	162
Creating Related Language Views.....	163
Understanding Related Language Views.....	163
Creating Views for One Base Table and One Related Language Table.....	164
Creating Views for Two Base Tables and One Related Language Table.....	165
Creating Views for Two Base Tables and Two Related Language Tables.....	166

Chapter 11

Working With Language-Sensitive Application Data.....	169
Understanding Data Editing in Related Language Tables and Base Tables.....	169
Editing Data in Multiple Languages.....	170
Understanding Multi-Language Entry.....	171
Enabling Multilanguage Entry.....	171
Entering Data in Multiple Languages.....	172

**Part 4
Translating PeopleSoft Applications**

Chapter 12

Modifying Terminology.....	177
Understanding Terminology Management	177
Terminology Management Overview.....	178
Terminology Searches.....	178
Terminology Replacement.....	179
Terminology Replacement Undo Process.....	179
Terminology Search Statuses.....	180
Prerequisites.....	181
Granting Terminology Search Privileges.....	181
Searching Terminology.....	182
Defining Search Criteria.....	182

- Cloning Existing Search Criteria.....187
- Running the Search Process.....188
- Viewing and Replacing Terminology Search Results.....188
 - Replacing Non-Message Text.....189
 - Replacing Message Text.....191
 - Running the Replace Process.....192
- Viewing and Undoing Terminology Replacement Results.....192
 - Undoing Data Replacements.....193
 - Undoing Text Replacements.....193
 - Running the Undo Process.....194
- Reviewing Terminology Search Information.....194
 - Reviewing Search Criteria.....195
 - Viewing Results for Search Only Searches.....195
 - Viewing Search Results for All Searches.....195
- Selecting Terminology Search Objects.....196
 - Viewing Searchable Objects.....196
 - Modifying Searchable Objects.....197
- Making Upgrade Considerations.....197

Chapter 13

- Adding New Languages.....199**
- Understanding the Addition of New Language Definitions.....199
- Adding New Language Codes to the System.....199
 - Determining PeopleSoft and ISO Codes for Your Language.....200
 - Determining Appropriate Non-Unicode Character Sets.....201
 - Adding New Translate Values to the LANGUAGE_CD Field.....202
 - Adding New Languages to the PSLANGUAGES Table.....203
 - Modifying Configuration Manager Windows Resources.....205
 - Modifying the PeopleSoft Pure Internet Architecture Signon Page.....206
 - Adding Text and Error Properties Files.....206
 - Adding Associated Images in the New Language.....207

Chapter 14

- Translating PeopleTools.....209**
- Understanding PeopleTools Translation.....209
- Prerequisites.....209
- Using Alternate Language DLLs.....210
- Locating Resource Directories.....211

Translating Resource Files.....	212
Translating Resource Files.....	212
Using MAKEALTL.BAT.....	213
Using MAKEUNIX.BAT.....	214
Translating Objects.....	215
Translating Properties Files.....	215
Translating Images.....	216

Chapter 15

Translating Application Definitions.....	217
Understanding Application Definition Translation.....	217
Using the Translation Designer Feature.....	217
Opening Translation Designer.....	218
Using Translation Designer Display Options.....	219
Working in Translation Designer.....	221
Using the Translation Pages.....	224
Understanding the Translation Pages.....	224
Translating Page Definitions.....	225
Translating Messages.....	226
Translating Menus.....	227
Translating Business Processes.....	230
Translating PeopleSoft Process Scheduler.....	232
Translating Mobile Data.....	234
Translating Portals.....	237
Translating Trees.....	238
Translating Records.....	240
Translating Queries.....	241
Translating Strings.....	242
Translating Application Engine Descriptions.....	243
Translating Review Page Text	244
Translating Application Data.....	245
Translating HTML Definitions.....	247

Appendix A

ISO Country and Currency Codes.....	249
ISO Country Codes.....	249
ISO Currency Codes.....	258

Glossary of PeopleSoft Terms.....269

Index285

About This PeopleBook

PeopleBooks provide you with the information that you need to implement and use PeopleSoft applications.

This preface discusses:

- PeopleSoft application prerequisites.
- PeopleSoft application fundamentals.
- Related documentation.
- Typographical conventions and visual cues.
- Comments and suggestions.
- Common elements in PeopleBooks.

Note. PeopleBooks document only page elements that require additional explanation. If a page element is not documented with the process or task in which it is used, then either it requires no additional explanation or it is documented with common elements for the section, chapter, PeopleBook, or product line. Elements that are common to all PeopleSoft applications are defined in this preface.

PeopleSoft Application Prerequisites

To benefit fully from the information that is covered in these books, you should have a basic understanding of how to use PeopleSoft applications.

See *Enterprise PeopleTools 8.45 PeopleBook: Using PeopleSoft Applications*.

You might also want to complete at least one PeopleSoft introductory training course.

You should be familiar with navigating the system and adding, updating, and deleting information by using PeopleSoft windows, menus, and pages. You should also be comfortable using the World Wide Web and the Microsoft Windows or Windows NT graphical user interface.

These books do not review navigation and other basics. They present the information that you need to use the system and implement your PeopleSoft applications most effectively.

PeopleSoft Application Fundamentals

Each application PeopleBook provides implementation and processing information for your PeopleSoft database. However, additional, essential information describing the setup and design of your system appears in a companion volume of documentation called the application fundamentals PeopleBook. Each PeopleSoft product line has its own version of this documentation.

The application fundamentals PeopleBook consists of important topics that apply to many or all PeopleSoft applications across a product line. Whether you are implementing a single application, some combination of applications within the product line, or the entire product line, you should be familiar with the contents of this central PeopleBook. It is the starting point for fundamentals, such as setting up control tables and administering security.

Related Documentation

This section discusses how to:

- Obtain documentation updates.
- Order printed documentation.

Obtaining Documentation Updates

You can find updates and additional documentation for this release, as well as previous releases, on the PeopleSoft Customer Connection website. Through the Documentation section of PeopleSoft Customer Connection, you can download files to add to your PeopleBook Library. You'll find a variety of useful and timely materials, including updates to the full PeopleSoft documentation that is delivered on your PeopleBooks CD-ROM.

Important! Before you upgrade, you must check PeopleSoft Customer Connection for updates to the upgrade instructions. PeopleSoft continually posts updates as the upgrade process is refined.

See Also

PeopleSoft Customer Connection, <https://www.peoplesoft.com/corp/en/login.jsp>

Ordering Printed Documentation

You can order printed, bound volumes of the complete PeopleSoft documentation that is delivered on your PeopleBooks CD-ROM. PeopleSoft makes printed documentation available for each major release shortly after the software is shipped. Customers and partners can order printed PeopleSoft documentation by using any of these methods:

- Web
- Telephone
- Email

Web

From the Documentation section of the PeopleSoft Customer Connection website, access the PeopleBooks Press website under the Ordering PeopleBooks topic. The PeopleBooks Press website is a joint venture between PeopleSoft and MMA Partners, the book print vendor. Use a credit card, money order, cashier's check, or purchase order to place your order.

Telephone

Contact MMA Partners at 877 588 2525.

Email

Send email to MMA Partners at peoplesoftpress@mmapartner.com.

See Also

PeopleSoft Customer Connection, <https://www.peoplesoft.com/corp/en/login.jsp>

Typographical Conventions and Visual Cues

This section discusses:

- Typographical conventions.
- Visual cues.
- Country, region, and industry identifiers.
- Currency codes.

Typographical Conventions

This table contains the typographical conventions that are used in PeopleBooks:

Typographical Convention or Visual Cue	Description
Bold	Indicates PeopleCode function names, method names, language constructs, and PeopleCode reserved words that must be included literally in the function call.
<i>Italics</i>	Indicates field values, emphasis, and PeopleSoft or other book-length publication titles. In PeopleCode syntax, italic items are placeholders for arguments that your program must supply. We also use italics when we refer to words as words or letters as letters, as in the following: Enter the letter <i>O</i> .
KEY+KEY	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press the W key.
Monospace font	Indicates a PeopleCode program or other code example.
“ ” (quotation marks)	Indicate chapter titles in cross-references and words that are used differently from their intended meanings.
. . . (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe ().

Typographical Convention or Visual Cue	Description
[] (square brackets)	Indicate optional items in PeopleCode syntax.
& (ampersand)	<p>When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object.</p> <p>Ampersands also precede all PeopleCode variables.</p>

Visual Cues

PeopleBooks contain the following visual cues.

Notes

Notes indicate information that you should pay particular attention to as you work with the PeopleSoft system.

Note. Example of a note.

If the note is preceded by *Important!*, the note is crucial and includes information that concerns what you must do for the system to function properly.

Important! Example of an important note.

Warnings

Warnings indicate crucial configuration considerations. Pay close attention to warning messages.

Warning! Example of a warning.

Cross-References

PeopleBooks provide cross-references either under the heading “See Also” or on a separate line preceded by the word *See*. Cross-references lead to other documentation that is pertinent to the immediately preceding documentation.

Country, Region, and Industry Identifiers

Information that applies only to a specific country, region, or industry is preceded by a standard identifier in parentheses. This identifier typically appears at the beginning of a section heading, but it may also appear at the beginning of a note or other text.

Example of a country-specific heading: “(FRA) Hiring an Employee”

Example of a region-specific heading: “(Latin America) Setting Up Depreciation”

Country Identifiers

Countries are identified with the International Organization for Standardization (ISO) country code.

See *About These PeopleBooks*, “ISO Country and Currency Codes,” ISO Country Codes.

Region Identifiers

Regions are identified by the region name. The following region identifiers may appear in PeopleBooks:

- Asia Pacific
- Europe
- Latin America
- North America

Industry Identifiers

Industries are identified by the industry name or by an abbreviation for that industry. The following industry identifiers may appear in PeopleBooks:

- USF (U.S. Federal)
- E&G (Education and Government)

Currency Codes

Monetary amounts are identified by the ISO currency code.

Appendix A, "ISO Country and Currency Codes" ISO Currency Codes.

Comments and Suggestions

Your comments are important to us. We encourage you to tell us what you like, or what you would like to see changed about PeopleBooks and other PeopleSoft reference and training materials. Please send your suggestions to:

PeopleSoft Product Documentation Manager PeopleSoft, Inc. 4460 Hacienda Drive Pleasanton, CA 94588

Or send email comments to doc@peoplesoft.com.

While we cannot guarantee to answer every email message, we will pay careful attention to your comments and suggestions.

Common Elements in These PeopleBooks

As of Date	The last date for which a report or process includes data.
Business Unit	An ID that represents a high-level organization of business information. You can use a business unit to define regional or departmental units within a larger organization.
Description	Enter up to 30 characters of text.
Effective Date	The date on which a table row becomes effective; the date that an action begins. For example, to close out a ledger on June 30, the effective date for the ledger closing would be July 1. This date also determines when

you can view and change the information. Pages or panels and batch processes that use the information use the current row.

Once, Always, and Don't Run

Select Once to run the request the next time the batch process runs. After the batch process runs, the process frequency is automatically set to Don't Run.

Select Always to run the request every time the batch process runs.

Select Don't Run to ignore the request when the batch process runs.

Report Manager

Click to access the Report List page, where you can view report content, check the status of a report, and see content detail messages (which show you a description of the report and the distribution list).

Process Monitor

Click to access the Process List page, where you can view the status of submitted process requests.

Run

Click to access the Process Scheduler request page, where you can specify the location where a process or job runs and the process output format.

Request ID

An ID that represents a set of selection criteria for a report or process.

User ID

An ID that represents the person who generates a transaction.

SetID

An ID that represents a set of control table information, or TableSets. TableSets enable you to share control table information and processing options among business units. The goal is to minimize redundant data and system maintenance tasks. When you assign a setID to a record group in a business unit, you indicate that all of the tables in the record group are shared between that business unit and any other business unit that also assigns that setID to that record group. For example, you can define a group of common job codes that are shared between several business units. Each business unit that shares the job codes is assigned the same setID for that record group.

Short Description

Enter up to 15 characters of text.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Process Scheduler

Enterprise PeopleTools 8.45 PeopleBook: Using PeopleSoft Applications

Global Technology Preface

This preface provides an overview of the contents of the *Enterprise PeopleTools 8.45 PeopleBook: Global Technology*.

PeopleTools Global Technology

This book covers PeopleTools features that are critical to a global implementation of PeopleSoft applications and is divided into four parts:

- **PART 1: Getting Started.** This part provides a quick overview of the setup and maintenance tasks that you may need to perform to implement your PeopleSoft applications for a multi-language, multinational environment.
- **PART 2: Configuring Your System for Global Implementation.** This part discusses some of the technological considerations you should take into account if you are implementing in multiple countries (even if you only use English language data).
- **PART 3: Working with PeopleSoft Applications in Multiple Languages.** This part discusses tasks to perform if you're using the non-English translations delivered by PeopleSoft and processing application data in multiple languages.
- **PART 4: Translating PeopleSoft Applications.** This section discusses tasks to perform if you need to modify delivered PeopleSoft terminology, translate objects into languages not supplied by PeopleSoft, translate products not translated by PeopleSoft, or translate your customizations.

PART 1

Getting Started

Chapter 1

Getting Started with Global Technology

CHAPTER 1

Getting Started with Global Technology

This chapter provides an overview of PeopleTools Global Technology and discusses how to implement PeopleTools in a multi-language, global environment.

PeopleTools Global Technology Overview

PeopleSoft applications are functionally equipped for implementation in global enterprises. However, country-specific application business logic is not enough to make an application global-ready. The technology behind the applications, PeopleTools, provides the core global functionality that is common to all PeopleSoft application products, including support for multiple languages in a single database, data display formats that use the standards that users expect with their language and country, and multiple time zone support. PeopleTools is designed so that a single implementation can serve users in different regions: users can share the same data while operating in different languages and applying different date/time and numeric formatting conventions.

This PeopleBook covers many aspects of global technology implementation, of interest to a wide variety of users:

- Administrators responsible for installing and maintaining the PeopleSoft database and applications.
- Developers designing new PeopleTools and PeopleSoft application objects.
- Translators translating PeopleTools and PeopleSoft application objects into new languages.
- End users setting up their user preferences or working with multiple languages.

PeopleTools global technology implementation can be divided into four main task areas:

1. Installation-level globalization tasks.

These tasks include selecting and setting up character sets and sort orders, installing PeopleSoft-delivered translations, and setting up currency and time zone options.

Many installation-level globalization tasks are required in a multinational implementation, even if you use only English-language data.

2. Post-installation globalization setup tasks.

These include a wide variety of setup tasks, some of them optional, that can be performed by administrators, developers, and end users.

3. Translating PeopleSoft applications.

PeopleSoft delivers non-English translations in a number of languages, which enables users to process data in multiple languages from a single database. PeopleSoft also provides tools that enable users to modify delivered translations to suit local needs.

You may also need to translate applications or application objects into languages that are not supplied by PeopleSoft, or you may need to translate your customizations. PeopleTools provides translation tools and recommendations for successful translations.

4. Developing global application objects.

Development for global applications requires special tools and special considerations. This book describes the PeopleTools that developers need and the practices that developers should follow to develop successful global, translatable applications.

PeopleTools Global Technology Implementation

Before using PeopleSoft applications in a multinational, multilanguage environment, you may need to perform some globalizing implementation tasks in addition to your standard PeopleSoft installation and implementation tasks.

Installation-Level Globalization Setup and Maintenance Tasks

Most installation tasks for a global implementation are covered in your PeopleSoft installation guide, but some global-only tasks are covered in this PeopleBook and are listed here. These tasks are performed primarily by system and database administrators. You can also perform some of these tasks after installation, if needed.

Step	Reference
Review the character set requirements for your database, application servers, and client workstations, and select an appropriate character set for your database.	See Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” page 35.
If required, convert your existing non-Unicode database to Unicode.	See Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” page 35. See also the special Unicode database conversion documentation available on PeopleSoft Customer Connection.
Set the database sort order. Most database systems require that you select a sort order when you create the database. Note. You should also set the PeopleTools in-memory sort options during installation, although you can change these after installation.	See Chapter 7, “Sorting in PeopleTools,” page 93.
Install the PeopleSoft-delivered translations that you have licensed, using the PeopleSoft Global Multi-Language CD-ROM. Note. If you are installing a new PeopleSoft database, we recommend that you install these translations at the same time that you install the English-language database. However, you may also install translations to existing PeopleSoft databases at any time.	See Chapter 10, “Using Related Language Tables,” Installing PeopleSoft-Provided Translations, page 155. Note. Multilanguage installation steps are covered in the PeopleSoft installation guide for your database platform. The PeopleTools Global Technology PeopleBook covers only installation steps for adding or upgrading additional licensed languages to an existing PeopleSoft database.
If the PeopleSoft database base language (language in which data is stored in the PeopleSoft core tables) is not to be English, swap the base language.	See Chapter 10, “Using Related Language Tables,” page 149.

Step	Reference
<p>Set up time zones and generate query offsets.</p> <p>Note. You can also maintain some time zone settings after installation.</p>	<p>See Chapter 3, “Setting and Maintaining Time Zones,” page 25.</p>
<p>If you plan to use currency conversion, set system-wide multicurrency options and resize currency fields, if necessary.</p> <p>Note. You can set currency settings at installation time or later.</p>	<p>See Chapter 5, “Controlling Currency Display Format.” Using System-Wide Multicurrency Settings, page 63.</p> <p>See Chapter 5, “Controlling Currency Display Format.” Resizing Currency Fields by Using the International Field Size Utility, page 65.</p>
<p>If you are running a Unicode database, convert any COBOL programs to a Unicode environment by running the COBOL conversion utility.</p>	<p>See Chapter 6, “Running COBOL in a Unicode Environment,” page 67.</p>

Post-installation Globalization Setup

After installing the PeopleSoft database, there are setup tasks that administrators, developers, or end users may need or want to perform to complete their global implementation. Most of the tasks in this list are optional, and depend on your organization’s particular needs.

Note. Some tasks listed above in the Installation-Level Globalization Setup and Maintenance Tasks section can also be performed after installation.

Step	Reference
<p>Set up language preferences for PeopleSoft Pure Internet Architecture display, your browser, and Microsoft Windows-based PeopleTools.</p> <p>Note. This setup is for language preferences only. Language preferences do not affect locale-specific formatting (such as number separators or date format).</p>	<p>See Chapter 2, “Controlling International Preferences.” Setting Up Language Preferences in the PeopleSoft Pure Internet Architecture, page 12.</p> <p>See Chapter 2, “Controlling International Preferences.” Applying Browser Language Preferences, page 15.</p> <p>See Chapter 2, “Controlling International Preferences.” Setting Up Language Preferences for Microsoft Windows-Based PeopleTools, page 17.</p>
<p>Set up PeopleSoft Pure Internet Architecture locale-based formatting, such as date format and number separators.</p> <p>You can set up locale-based formatting defaults at three levels: user, locale, and system-wide. The user-defined defaults override the locale-specific defaults, which override the system-wide defaults.</p> <p>User-defined defaults can be set up by end users. Locale-specific and system-wide defaults should be performed by a systems administrator or similar role.</p> <p>This task is necessary only if you need to override the delivered formatting.</p>	<p>See Chapter 2, “Controlling International Preferences.” Setting Up PeopleSoft Pure Internet Architecture Locale-Based Formatting, page 18.</p>

Step	Reference
Enable multilanguage data entry for users who will enter data in multiple languages during a single session. This task can be performed by end users or administrators.	See Chapter 11, “Working With Language-Sensitive Application Data,” Editing Data in Multiple Languages, page 170.
If you use Microsoft Excel 97, select language-sensitive preferences for downloads from PeopleSoft pages and queries to Microsoft Excel.	See Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” International Versions of Microsoft Excel, page 141.

Translation Tasks

If you want to modify PeopleSoft-delivered terminology in any delivered language that you use, including English, perform the following task:

Step	Reference
Modify PeopleSoft-delivered terminology and translations to meet local requirements.	See Chapter 12, “Modifying Terminology,” page 177.

If you have developed new application objects, or need to translate PeopleSoft-delivered PeopleTools and application objects into languages that PeopleSoft does not deliver, you may need to perform the following tasks.

Step	Reference
Define any languages you need that are not delivered by PeopleSoft.	See Chapter 13, “Adding New Languages,” page 199.
Translate PeopleTools objects into non-delivered languages or modify PeopleSoft-delivered translations of PeopleTools objects.	See Chapter 14, “Translating PeopleTools,” page 209.
Translate PeopleSoft application definitions into non-delivered languages (or modify delivered PeopleSoft translations) using the Translation Designer feature, Translation pages, and PeopleSoft Application Designer.	See Chapter 15, “Translating Application Definitions,” page 217.

Global Application Development Tasks

If you are developing new application objects for use in a global implementation, you may need to perform the following tasks.

Step	Reference
Control time zone display.	See Chapter 3, “Setting and Maintaining Time Zones,” page 25.
Validate field lengths for the character sets you will use.	See Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” Understanding Character Sets, page 35. See Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” Validating Field Length, page 55.

Step	Reference
Design appropriate field formats for character sets you will use.	See Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” Handling International Characters in PeopleTools, page 58.
Set up currency amount and currency control fields.	See Chapter 5, “Controlling Currency Display Format,” page 61.
Use appropriate sorts in the objects that you develop.	See Chapter 7, “Sorting in PeopleTools,” page 93.
Design reports and configure reporting tools for global requirements.	See Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” page 101. See also discussions of language- and locale-specific formatting considerations in the relevant reporting tools PeopleBooks.
Create related language tables and views for any language-sensitive tables and views that you create or modify.	See Chapter 10, “Using Related Language Tables,” page 149.

We also provide a discussion of best practices for developing easily translatable applications.

See [Chapter 9, “Developing Global Applications,” page 143.](#)

PeopleTools delivers a number of globalization-related PeopleCode functions.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference*, “PeopleCode Built-in Functions”.

Other Sources of Information

In the implementation planning phase, take advantage of all PeopleSoft sources of information, including the installation and upgrade guides, release notes, red papers, updates and fixes posted to PeopleSoft Customer Connection, PeopleSoft curriculum guides, and other PeopleBooks.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Getting Started with Enterprise PeopleTools

PART 2

Configuring Your System for Global Implementation

Chapter 2
Controlling International Preferences

Chapter 3
Setting and Maintaining Time Zones

Chapter 4
Selecting and Configuring Character Sets and Language Input and Output

Chapter 5
Controlling Currency Display Format

Chapter 6
Running COBOL in a Unicode Environment

Chapter 7
Sorting in PeopleTools

Chapter 8
Understanding Global Reporting and Data Analysis Tools

Chapter 9
Developing Global Applications

CHAPTER 2

Controlling International Preferences

This chapter provides an overview of international preferences and discusses how to:

- Set up language preferences in the PeopleSoft Pure Internet Architecture.
- Apply browser language preferences.
- Set up language preferences for Microsoft Windows-based PeopleTools.
- Set up locale-based formatting for the PeopleSoft Pure Internet Architecture.
- Set up locale-based formatting for Microsoft Windows-based PeopleTools.

Understanding International Preferences

The international preference settings control the language in which users view and use the PeopleSoft system, as well as locale formatting conventions for dates, times, and numbers. The PeopleSoft system treats language preferences and local formatting preferences separately.

Language Preferences

You can set up language preferences for your internet browser, for the PeopleSoft Pure Internet Architecture (by session and by user profile), and for the PeopleSoft Windows-based client. Your user, or session language, helps to determine the language of your PeopleSoft interface, such as field and menu labels, as well as the language of the data that you maintain or view during the session. The internet browser language setting helps to determine the language of your PeopleSoft signon page in the PeopleSoft Pure Internet Architecture.

Locale-Based Formatting Preferences

Locale-based formatting determines such preferences as date, time, and number formatting. You can set up locale-based formatting preferences at three levels, each of which overrides the one above it:

1. System-wide
2. Locale-specific
3. User-specific

You can set these formatting preferences for both the PeopleSoft Pure Internet Architecture and for the Windows client PeopleTools. The preferences you set up for one is independent of those you set up for the other.

Setting Up Language Preferences in the PeopleSoft Pure Internet Architecture

This section provides an overview of the session language and discusses how to:

- Choose a session language.
- Change the session language while signed in.
- Change the session language programmatically.
- Determine the session language when the sign-in screen is bypassed.
- Control the user profile language preference.
- Determine the language of help documentation.

See Also

[Chapter 10, “Using Related Language Tables,” page 149](#)

Understanding the Session Language

In the PeopleSoft Pure Internet Architecture, the *session language* is the language selected by the user on the signon page before entering a user name and password. The session language controls the language in which the user views language-sensitive data and objects in the application for the duration of that session.

If the session language is the database’s base language, then all language-sensitive data, messages, and objects appear in the base language. If the session language is a non-base language, then any language-sensitive data, messages, and objects that have been translated appear in the preferred language; those that have not been translated appear in the base language. For example, if a user’s language code for a particular session is Spanish, PeopleTools attempts to display menus and field labels in Spanish for each page viewed in that session, and if the table from which the data is being fetched has a related language record associated with it, PeopleTools also attempts to fetch the Spanish translations of the data from this table.

Choosing a Session Language

Users choose a session language when they sign in to the system. Users can change their session language during the PeopleSoft session; however, this is typically done only for development or testing purposes. You can also program applications to change session languages, so that your application has its own language selection options.

When the sign-in page initially appears, the page elements on that page appear in the language designated in the user’s browser language settings.

To choose a different session language, click one of the available language links before signing in. Clicking the link updates the sign-in screen to reflect the new language choice, and signing in using the updated screen confirms the new session language. Only languages that have been installed and enabled by the system administrator can be used to sign in to the database. If you select a language that has not been enabled in the database to which you are logging in, you receive an error message.

To enable and disable languages for signing in to the database, use the Maintain Languages page. To access this page, select PeopleTools, Utilities, Maintain Languages.

If you will not be using all languages delivered by PeopleSoft, or if you need to add a new language to the system, you can modify the sign-in page to add or remove language links. Remember, the base language of your system should always be available.

See Also

[Chapter 13, “Adding New Languages,” page 199](#)

Changing the Session Language While Signed In

During development, testing, or debugging, you might want to change the language of your session without signing out of the PeopleSoft system.

Use the International Preferences page to change your session language during a session. Any change made using this page is valid only for the remaining duration of the current session.

To change your session language without signing out:

1. Select PeopleTools, Utilities, International, Preferences.
The International Preferences page appears.
2. Select a language in the Language Preference field.
3. Save the page.

Changes take effect when a new component is loaded.

Changing the Session Language Programmatically

You can change user language preferences programmatically for the remainder of a PeopleSoft Pure Internet Architecture session. This feature is useful if you are building a front end to your PeopleSoft system that does not involve the sign-in screen, and if you want to enable users to change their language preference at any time.

For example, an internet store application might enable anonymous sign-in by customers of the store to browse a catalog, and therefore would not display the sign-in screen. The PeopleCode SetLanguage built-in function could be linked to a PeopleCode program behind a button on a page to enable users to change the language of the interface after they have already been signed in as guest users.

The SetLanguage function temporarily overrides the user’s default language preference and the *%Language* system variable, which evaluates to the language code for the current preferred language.

Calling the SetLanguage function is the same as changing the Language Preference field on the International Preferences page.

The *%Language_Base* system variable makes it simple to determine the current preferred language and build language-specific functionality in PeopleCode.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “PeopleCode Built-in Functions,” SetLanguage

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “System Variables,” %Language

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “System Variables,” %Language_Base

Determining the Session Language When the Signon Page is Bypassed

PeopleSoft offers the following ways to sign in to a PeopleSoft session without seeing the signon page:

- Direct access.

You can configure the web server to enable users to access the PeopleSoft system without having to sign in. In these cases, users access the system using a predetermined user ID.

The language preference for users who access the system this way is determined by the user profile language preference for that user ID.

- When bypassing the signon page, you can override the user profile language preference by explicitly setting the language code within a direct query string using the parameter `&languageCd=<target language>`
- Single signon.

You need to select a language preference only once during the single signon.

All subsequent PeopleSoft sessions that are controlled by the single signon use the same language preference.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Security Administration, “Setting up Digital Certificates and Single Signon”

Controlling the User Profile Language Preference

In addition to the session language preference that is selected at sign-in time and is valid for the current session only, the PeopleSoft Pure Internet Architecture also maintains a language preference on each user’s profile. This language preference is used to determine the user’s preferred language when that user is not signed into the PeopleSoft system. For example, when PeopleSoft Process Scheduler runs a Structured Query Language report (SQR), or if the workflow system sends an email to the user, the user language preference is used.

In these cases, the user might not be signed into the PeopleSoft system, or another user might have initiated the transaction, so there is no session language that can be used. For example, a manager might be sent an email by the workflow system requesting approval for a purchase requisition.

To set the language preference for a user, use the User Profiles page. To change your own language preference, use the simplified General Profile Information page.

To set the user profile language preference for another user:

1. Select PeopleTools, Security, User Profiles, User Profiles.
The User Profile search page appears.
2. Use the standard search method to choose a user ID, and click the Search button.
The User Profiles - General page appears.
3. From the Language Code drop-down list, select a language.
4. Save the page.

To set your user profile language preference:

1. From the menu pagelet, select My System Profile.
The General Profile Information page appears.

2. In the Personalization section, choose a language in the My preferred language for reports and email is drop-down list .
3. Save the page.

See Also

[Chapter 11, “Working With Language-Sensitive Application Data,” page 169](#)

Determining the Language of Help Documentation

If a user accesses a document from a PeopleSoft application by clicking Help in a PeopleSoft page or by pressing F1 from the Microsoft Windows client, the system opens the version of the document that corresponds to the session language. If the system doesn't find a version of the documentation in the appropriate language, it brings up the English documentation, if available.

Applying Browser Language Preferences

This section provides an overview of browser language preferences and discusses how to:

- Set browser language preferences.
- Use browser language settings to determine the sign-in language.

Understanding Browser Language Preferences

Web browsers enable users to specify a list of preferred languages for web content. PeopleTools uses the first language listed in the browser's preferred language list to determine the default language of the PeopleSoft signon page. If the user's preferred language is not available, the default sign-in screen is displayed in U.S. English (en-US).

Setting Browser Language Preferences

In Microsoft Internet Explorer, the preferred language list is located in the Language Preferences dialog box. You can add new languages using the Add Language dialog box.

The representation of language codes in the browser's preferred language list uses two-to-five character ISO standard locale codes made up of a language and country portion. For example *es* represents Spanish and *ar* represents Argentina. A two-letter code represents only a language without reference to a specific country or territory.

Netscape and other browsers have a similar dialog box that enables you to create an ordered list of language preferences.

Using Browser Language Settings to Determine the Sign-in Language

For PeopleTools to display the sign-in page in the user's preferred language as determined from browser settings, the language codes used by the browser are mapped to PeopleSoft three-letter language codes. The PeopleSoft system reads the PSLANGUAGES table to perform this mapping.

As delivered, the PeopleSoft system includes the following mappings:

PeopleSoft Language Code	Description	Corresponding ISO Locales
CFR	Canadian French	fr_CA
DAN	Danish	da
DUT	Dutch	nl
ENG	English	en
ESP	Spanish	es
FRA	French	fr
GER	German	de
GRK	Greek	el
ITA	Italian	it
JPN	Japanese	ja
KOR	Korean	ko
POL	Polish	pl
POR	Portuguese	pt
SVE	Swedish	sv
THA	Thai	th
ZHS	Simplified Chinese	zh_CN
ZHT	Traditional Chinese	zh_HK, zh_TW

The two-part ISO locale provides flexibility when languages vary from country to country. For example, the ISO code for French is *fr*: French is spoken in several countries, each with its own two-character country code, including France (FR), Belgium (BE), Canada (CA), and Switzerland (CH). The full locale code consists of a lowercase language code, an underscore, and an uppercase country code. So the code for Canadian French is *fr_CA*.

In many cases, the country distinction between languages is not relevant, or PeopleSoft does not deliver a country-specific translation. In these cases, many individual ISO locale codes may be mapped to a single PeopleSoft language code. For example, PeopleSoft delivers a Canadian French translation with the PeopleSoft language code of CFR. Therefore, the ISO locale of fr_CA is mapped to the PeopleSoft language code of CFR. However, PeopleSoft does not provide a Belgian French or Swiss French translation, so the fr_BE and fr_CH ISO locales fall back to fr, and are both mapped to the FRA PeopleSoft language code.

Setting Up Language Preferences for Microsoft Windows-Based PeopleTools

This section provides an overview of language preferences in PeopleTools for Microsoft Windows and discusses how to control PeopleSoft Configuration Manager language settings.

Understanding Language Preferences for Microsoft Windows-Based PeopleTools

In the Microsoft Windows-based PeopleTools, the language setting within PeopleSoft Configuration Manager controls the language in which the user views almost all language-sensitive data and objects in the application.

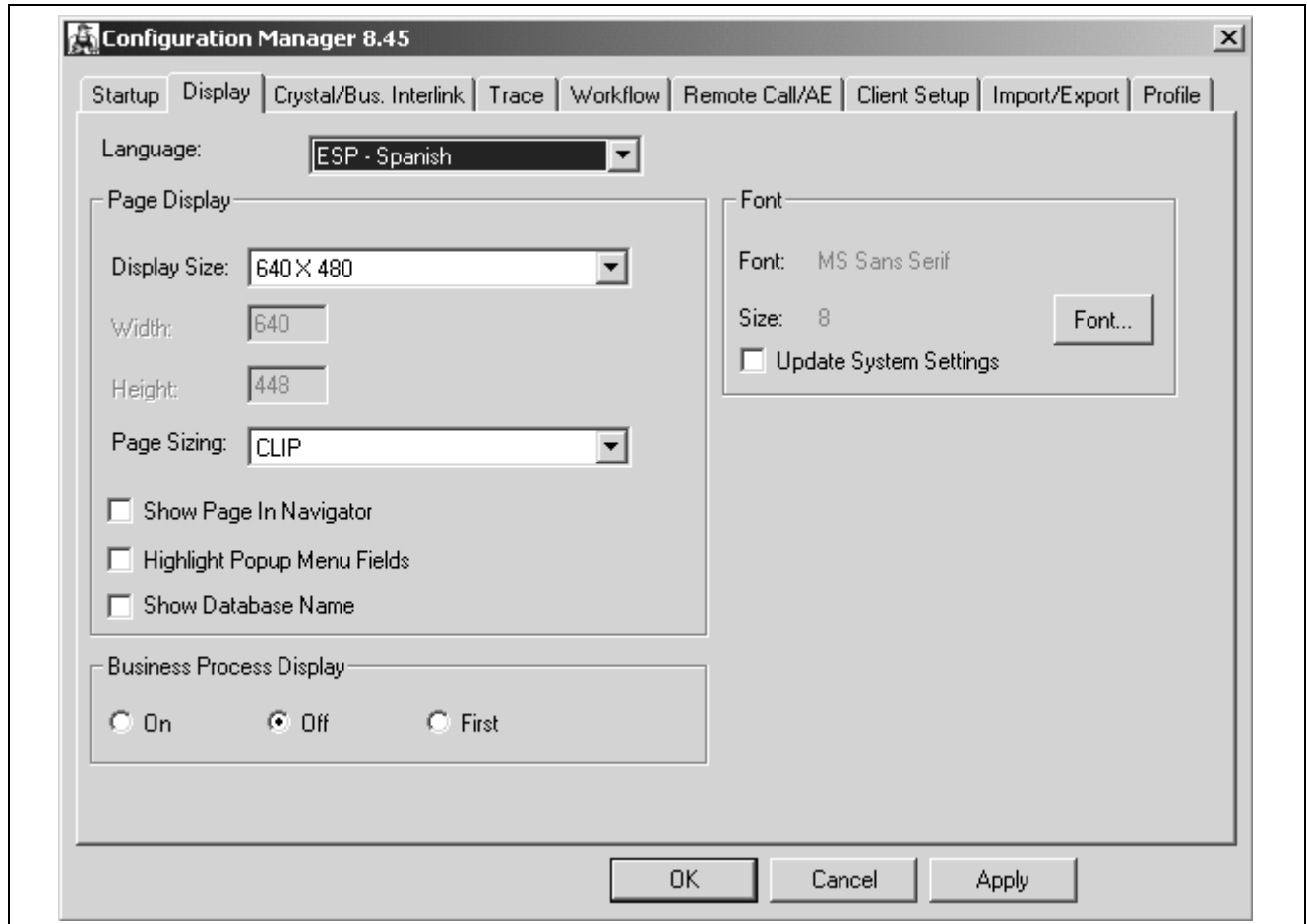
Just like in the PeopleSoft Pure Internet Architecture, if the language selected in PeopleSoft Configuration Manager is the base language, then all language-sensitive data, messages, and objects appear in the base language. If the language in PeopleSoft Configuration Manager is a non-base language, then any language-sensitive data, messages, and objects that have been translated appear to the user in the preferred language; those that have not been translated appear in the base language.

See Also

[Chapter 10, “Using Related Language Tables,” page 149](#)

Controlling PeopleSoft Configuration Manager Language Settings

The PeopleSoft Configuration Manager language setting is maintained on each Microsoft Windows client workstation and determines the language in which the user views and maintains almost all language-sensitive data and objects in the Microsoft Windows-based PeopleTools.



Configuration Manager dialog box: Display tab

To set the PeopleSoft Configuration Manager language:

1. Start PeopleSoft Configuration Manager.
2. Select the Display tab.
3. Select a language in the Language field.
4. Click OK.

Setting Up PeopleSoft Pure Internet Architecture Locale-Based Formatting

This section provides an overview of locale-based formatting drivers and discusses how to:

- Define user-specific formatting.
- Define locales for locale-specific formatting.
- Define locale-specific formatting.
- Apply system-wide default formatting.

See Also

[Chapter 5, “Controlling Currency Display Format,” page 61](#)

Enterprise PeopleTools 8.45 PeopleBook: Security Administration, “Managing PeopleSoft Personalizations”

Understanding Locale-Based Formatting Drivers

The locale-based personalization architecture in the PeopleSoft Pure Internet Architecture provides a flexible framework for formatting locale-sensitive data. Three levels of personalization settings drive the formatting of locale-based data in the PeopleSoft Pure Internet Architecture:

- User-specific.
- Locale-specific.
- System-wide.

These levels ensure that each user can specify how to format locale-based data, such as numbers, dates and times, while maintaining an intelligent set of default formats for inexpert users and users who don't have specific preferences.

When determining how to format these types of data, PeopleTools checks for formatting preferences at each of these levels in the order of user-specific, locale-specific, and finally system-wide.

For example, a first-time user may sign in to the PeopleSoft Pure Internet Architecture as a French user. Because the user is a first-time user of the system, PeopleTools does not have any personalization information specific to that user. Next, PeopleTools checks the personalizations tables to see if a locale-specific formatting setting exists for the French language. If locale-specific settings for French users are found, the system formats all numbers, dates, and times for that session using the settings defined by the system administrator for French users.

However, it is possible that the user, even though signed in using the French language, might prefer another type of formatting for number separators. To register a user-specific preference, the user opens the My Personalizations - Personalizations page, accesses personalization options for regional settings, and overrides the Digit Group Separator field value. Whenever that user accesses the PeopleSoft Pure Internet Architecture, the decimal separator is a custom value, regardless of language choice. All other preferences are read from the locale-specific personalizations.

If a user signs in to the PeopleSoft Pure Internet Architecture using a language for which locale-specific personalizations have not been set up by the system administrator, PeopleTools uses system-wide defaults.

Using the personalizations system in PeopleTools, the system administrator can restrict which personalization options are driven by a locale and which can be overridden. In certain organizations where a corporate-wide policy exists for formatting numbers, dates, or times, the system administrator might disable the locale sensitivity or user overrides for locale-sensitive data formats. In this case, the system-wide defaults take effect for each personalization option disabled by the system administrator.

You use the following pages to manage locale-based personalization options:

Page Name	Navigation	Usage
Locale Definition	PeopleTools, Personalization, Locales	Control the locales for which you can specify defaults.
Locale Defaults	PeopleTools, Personalization, Locale Defaults	Specify Personalization Option defaults for locales appearing on the Locale Definition page.

Pages Used to Set Up PeopleSoft Pure Internet Architecture Locale-Based Formatting

Page Name	Object Name	Navigation	Usage
My Personalizations - Personalizations (Option Category: Regional Settings)	PSUSERSELFPRSNL	My Personalizations. Click the Personalize Option button for <i>Regional Settings</i> .	End users access this page to view and modify their personal locale-based formatting preferences.
Locale Definition	PSLOCALEDEFN	PeopleTools, Personalization, Locales	Control the locales for which you can specify defaults.
Locale Defaults	PSLOCALEOPTNDFLTS	PeopleTools, Personalization, Locale Defaults	Specify personalization option defaults for locales appearing on the Locale Definition page. Locale-based personalizations enable PeopleTools to intelligently determine appropriate data formatting based on the language of the user logging in to PeopleTools if they have not specified individual user-based overrides for each formatting option.
Define Personalizations	PSUSEROPTNDEFN	PeopleTools, Personalization, Personalization Options	Set up system-wide locale-based formatting defaults. These defaults are applied to a user session if no locale-specific defaults or user-specific defaults have been specified for an option.

Defining User-Specific Formatting

Access the My Personalizations - Personalizations page.

Option Category: Regional Settings

Personalizations			Find	First	1-10 of 10	Last
Personalization Option	Default Value	Override Value				
Afternoon designator (PM, pm)	PM	<input type="text"/>				Explain
Decimal Separator	.	<input type="text"/>				Explain
Date Format	MMDDYY	<input type="text"/>				Explain
Date Separator	/	<input type="text"/>				Explain
Local Time Zone	Pacific Time (US), Tijuana	<input type="text"/>				Explain
Morning designator (AM, am)	AM	<input type="text"/>				Explain
Time Format	12 hour clock	<input type="text"/>				Explain
Time Separator	:	<input type="text"/>				Explain
Digit Group Separator	,	<input type="text"/>				Explain
Use Local Timezone	No	<input type="text"/>				Explain

My Personalizations - Personalizations page: Regional Settings

Use this page to specify how PeopleTools should format each field that contains numbers, dates, and times. Each individual formatting element can be overridden on a user-by-user basis.

To override any or all of these values, enter the appropriate value or select one from the drop-down list in the Override Value column. For example, if you prefer to use a dash (-) instead of a slash (/) to separate the components of dates, simply enter a dash character in the Override Value column for the Date Separator personalization option. Values maintained here are specific to your user ID.

If you leave any of these override values blank, they will be evaluated next time you sign on based on your language code, and if locale-specific preferences aren't found for your language, they will be defaulted to the system-wide values as shown in the Default Value column on this page.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Security Administration, “Managing PeopleSoft Personalizations,” Working with the My Personalizations Interface

Enterprise PeopleTools 8.45 PeopleBook: Security Administration, “Managing PeopleSoft Personalizations,” Understanding International and Regional Options

Defining Locales for Locale-Specific Formatting

Access the Locale Definition page.

Locale Definition

Customize Find View 100 First 1-25 of 142 Last		
*Locale Code	*Description	
af	Afrikaans	+ -
ar	Arabic	+ -
ar-ae	Arabic (U.A.E.)	+ -
ar-bh	Arabic (Bahrain)	+ -
ar-dz	Arabic (Algeria)	+ -
ar-eg	Arabic (Egypt)	+ -
ar-iq	Arabic (Iraq)	+ -
ar-jo	Arabic (Jordan)	+ -
ar-kw	Arabic (Kuwait)	+ -
ar-lb	Arabic (Lebanon)	+ -
ar-ly	Arabic (Libya)	+ -

Local Definition page

Use this page to add locales that can have defaults specified on the Locale Defaults page. These locales are different from PeopleSoft language codes; they typically contain both language and country information. When formatting data fields, both these elements are typically needed, given that many languages are spoken in different countries, with each country often following different formatting conventions. Having only information on the user's language is not sufficient to determine even a default level of locale-sensitive data formatting.

For example, English is spoken in several countries that radically disagree on date formatting. In the United States, dates are typically formatted as month, day, year, whereas in the United Kingdom and elsewhere, they are typically formatted as day, month, year. Therefore if PeopleTools only has a sign-in language of English, and no country information, it could not determine which of the two possible date formats to use.

The session language code cannot be used to determine the specific locale for a session, due to its lack of country information. Instead, the browser's language preference list is read, and the top (the most preferred) locale in that list is used to look up locale-specific formats in the Personalizations Locale Definitions page. Therefore, you should ensure that each browser locale used by your system's users appears in the Locale Definition page.

Assigning Locale-Specific Formatting

Access the Locale Defaults page.

Locale Defaults				
Customize Find View 100 First 1-25 of 670 Last				
*Option Category Level	*User Option	*Locale Code	Description	*Override Value
Tools	Afternoon designator (PM, prr)	af	Afrikaans	nm
Tools	Afternoon designator (PM, prr)	en	English	PM
Tools	Afternoon designator (PM, prr)	en-au	English (Australia)	PM
Tools	Afternoon designator (PM, prr)	en-bz	English (Belize)	PM
Tools	Afternoon designator (PM, prr)	en-ca	English (Canada)	PM
Tools	Afternoon designator (PM, prr)	en-nz	English (New Zealand)	PM
Tools	Afternoon designator (PM, prr)	en-ph	English (Philippines)	PM
Tools	Afternoon designator (PM, prr)	en-tt	English (Trinidad)	PM

Local Defaults page

Use this page to assign default values for personalization options based on locale code.

To more easily view defaults for a particular locale, click the Locale Code column heading to sort the data on this page by the locale code.

Note. For some international locales, a space character is used as a digit group separator (for example, thousands). However, the PeopleSoft Pure Internet Architecture interprets a single space in a field as a null value. To use the space character as a digit group separator, define the override value as a space between single quotation marks. This personalization can be defined at the system-wide, locale-default, or individual user level. This special treatment of the space character currently only takes effect for the digit group separator user option.

Applying System-Wide Default Formatting

System-wide formatting is applied for the session only if a particular personalization option can't be found that is specific to the user or the browser's locale. System-wide defaults are defined using the Define Personalizations page. To apply system-wide default formatting, select PeopleTools, Personalization, Personalization Options, and click the Format tab.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Security Administration, "Managing PeopleSoft Personalizations"

Setting up Locale-Based Formatting for Microsoft Windows-based PeopleTools

The Microsoft Windows Regional Options dialog box (accessed through the Microsoft Windows control panel) enables you to choose the format for the display of numbers, dates, times, and currencies while operating in PeopleSoft Application Designer and other Microsoft Windows components of PeopleTools.

Note. Regional Options that you set up in Microsoft Windows do not effect the display or maintenance of data in the PeopleSoft Pure Internet Architecture.

Currency formatting is controlled by PeopleTools and is independent of the Regional Options dialog box.

See Also

Chapter 5, “Controlling Currency Display Format,” page 61

CHAPTER 3

Setting and Maintaining Time Zones

This chapter provides an overview of time zones and discusses how to:

- Set the base time zone.
- Maintain time zones.
- Control the time zone display when you develop PeopleSoft application objects.

Understanding Time Zones

PeopleSoft applications store times based on a system-wide base time zone and support the display of times relative to a user's local time zone or relative to the time zone in which a transaction is entered (called a specified time zone).

For example, if the base time zone is universal coordinated time (UTC, also known as Greenwich mean time or GMT), a time entered as 10 a.m. in U.S. eastern standard time (EST) is stored as 3 p.m., because of the time difference between the two locations. This allows for easy comparison and manipulation of times, so that PeopleCode developers do not have to worry about time zone differences. Additionally, all batch processes, such as Structured Query Reports and COBOL, operate in the base time zone of the system.

However, although the time is stored as 3 p.m., it can still appear to the user as 10 a.m. EST, or as the appropriate time for any time zone that you choose. You can also set up the system to enable users to choose the time zone for specific time or datetime displays.

To support display in alternate time zones, the PeopleSoft system maintains a global list of time zones, including information about daylight savings time observances. When a time zone begins or ends in daylight savings time, both the description of the zone and the times that are associated with that zone are adjusted to reflect the change. This approach ensures chronological consistency throughout an organization's geographically dispersed locations, while allowing for flexibility in how users see times.

Base Time Zone Details

This time zone is used for display when no other time zone is specified. The base time zone must match the time zone in which the database server is operating, specifically the time zone to which the meta-SQL %CurrentDateTime token returns. For example, in an Oracle database, the base time zone is the time zone in which the *SYSDATE* system variable is returned. You must enter this time zone on the PeopleTools Options page for your system to function correctly.

The base time zone is also used for the following:

- All effective-date processing.

Rows become effective when midnight passes on the effective date according to the base time zone of the system.

- Internal processing in PeopleCode and all batch processes.
- Display on pages and in reports, if no other time zone information is specified or available.

User Time Zone Details

The user time zone is read from the operating system of the user's web browser in the PeopleSoft Pure Internet Architecture.

Specified Time Zone Details

Specified time zones are useful for applications where users must see the same time and time zone as those who enter transactions, such as in call processing systems. To implement a specified time zone field, your record must contain a time zone control field.

Setting the Base Time Zone

Select PeopleTools, Utilities, Administration, PeopleTools Options to access the PeopleTools Options page.

PeopleTools Options

Environment Long Name: Environment Short Name:

System Type:

Language Settings

Language Code: English *Sort Order Option: Translations Change Last Update

General Options

Background Disconnect Interval: Temp Table Instances (Total):

Multi-Company Organization Temp Table Instances (Online):

Multi-Currency *Maximum App Message Size:

Use Business Unit in nVision Base Time Zone:

Use Secure Rep Rqst in nVision Last Help Context # Used:

Multiple Jobs Allowed *Data Field Length Checking:

Allow DB Optimizer Trace *Maximum Attachment Chunk Size:

Grant Access

Platform Compatibility Mode

Case Insensitive Searching

Allow NT batch when CCSID<>37 Upgrade Project Commit Limit:

Style Sheet Name: Upgrade Project Commit Limit:

Branding Application Package:

Branding Application Class:

Tree Manager Options

Use Tree Update Reservation

Max Tree Inactivity Period,min:

Help Options

F1 Help URL:

Ctrl-F1 Help URL:

PeopleTools Options page

The base time zone that you define for PeopleTools must match the time zone used by your database. Any discrepancy between the two leads to inaccurate time processing.

To set the base time zone:

1. Select a time zone from the Base Time Zone field.
 Values are maintained in the Time Zones component (TIMEZONEPNLGRP).
 See [Chapter 3, “Setting and Maintaining Time Zones,” Maintaining Time Zones, page 28.](#)
2. Click the Save button.

Note. After selecting a base time zone during installation, avoid changing it. There is no automated processing to translate existing time information in the database to a new base time zone, and if you change the time zone defined on the PeopleTools Options page, no adjustment is made to existing time data in your database.

Maintaining Time Zones

This section provides an overview of time zone maintenance and discusses how to:

- View time zone definitions.
- Generate time zone offsets.
- View daylight savings start and end dates.

Understanding Time Zone Maintenance

PeopleSoft delivers a comprehensive list of time zones, including information about daylight savings time observances. You can update the data if necessary, using the Time Zones component. For example, you can include a time zone for a locality with daylight savings observances that differ from other parts of its time zone.

Viewing Time Zone Definitions

Select PeopleTools, Utilities, International, Time Zones to access the Time Zone Data page.

Time Zone	Time Zone Description	Time Zone Standard Label	Time Zone Daylight Label	Universal Time Coord Offset
HST	Hawaiian Time	HST		-600
AWST	Australian Western Standard Time	AWST	AWDT	480
JST	Japan Standard Time	JST	JST	540
UTC	Universal Time Coordinated, Zulu	UTC		
EST	Eastern Time (US)	EST	EDT	-300
MST	Mountain Time (US)	MST	MDT	-420
GMT	Greenwich Mean Time	GMT	BST	
AKST	Alaska Time (US)	AKST	AKDT	-540
AST	Atlantic Time (Canada)	AST	ADT	-240
PST	Pacific Time (US), Tijuana	PST	PDT	-480

Time Zone Data page: Time Zones tab

Time Zone Tab

Time Zone Standard Label Displays the label that appears for the time zone during standard time.

Time Zone Daylight Label Displays the label that appears for the time zone during daylight savings time.

Universal Coord Time Offset (universal coordinated time offset)

Displays the number of minutes the time is offset from UTC.

A positive offset indicates a time zone east of UTC; a negative offset indicates a time zone west of UTC.

For example, the time zone for India, which is 5 and a half hours east of UTC, has an offset of +330. U.S. pacific standard time (PST), which is 8 hours west of UTC, has an offset of -480.

Generate Query Offsets

Click to access the open the Time Zone Offset Generation page. On this page, you can use the information from the Time Zone Data page to populate the PSTZOFFSET table with offsets for all the time zones and their daylight savings time periods for a specified period of time. This makes the time zone information available in a format that can be easily accessed with SQL. This may be useful for PeopleSoft Query. The PeopleSoft system does not require that you populate this table; you need to do so only if you require access to this information. If you do use the information, regenerate the offsets anytime the underlying time zone data is changed.

Daylight Savings Tab

Select the Daylight Savings tab to view daylight savings offset information.

*Time Zone	Time Zone Description	Observes Daylight Savings Time	Daylight Savings Offset	Daylight Saving Time Start	Daylight Savings Time End
HST	Hawaiian Time	<input type="checkbox"/>			
AWST	Australian Western Standard Time	<input type="checkbox"/>			
JST	Japan Standard Time	<input type="checkbox"/>	60		
UTC	Universal Time Coordinated, Zulu	<input type="checkbox"/>			
EST	Eastern Time (US)	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct
MST	Mountain Time (US)	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct
GMT	Greenwich Mean Time	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct
AKST	Alaska Time (US)	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct
AST	Atlantic Time (Canada)	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct
PST	Pacific Time (US), Tijuana	<input checked="" type="checkbox"/>	60	2FirstSunApr	2LastSunOct

Time Zone Data page: Daylight Savings tab

Daylight Savings Offset

Displays the number of minutes by which the time is offset during daylight savings time.

Daylight Savings Time Start

Displays the day that daylight savings time begins. This field prompts against values from the Daylight Savings page.

Daylight Savings Time End

Displays the day that daylight savings time ends. This field prompts against values from the Daylight Savings page.

Generating Time Zone Offsets

Click the Generate Query Offsets button on the Time Zone Data page to access the Time Zone Offset Generation page.

To generate query offsets:

1. Enter a start date and an end date.

Select a reasonable range of dates. With the time zones delivered with the PeopleSoft system, more than 100 rows are generated for each year.

2. Click OK.

The data is generated for the table. Any existing data for the same range of dates is overwritten.

For example, the following data from the PSTZOFFSET table represents one time zone, EST, for a two-year period, from January 1, 1999 to January 1, 2001. It is based on a system with PST as the base time zone. When you generate query offsets, the PSZOFFSET table contains similar data for all the time zones.

TIMEZONE	STARTDATETIME	ENDDATETIME	BASEOFFSET	TIMEZONELABEL
EST	1999-01-01 00:00:00.000	1999-04-03 23:00:00.000	180	EST
EST	1999-04-03 23:00:00.000	1999-04-04 02:00:00.000	240	EDT
EST	1999-04-04 02:00:00.000	1999-10-30 23:00:00.000	180	EDT
EST	1999-10-30 23:00:00.000	1999-10-31 02:00:00.000	120	EST
EST	1999-10-31 02:00:00.000	2000-04-01 23:00:00.000	180	EST
EST	2000-04-01 23:00:00.000	2000-04-02 02:00:00.000	240	EDT
EST	2000-04-02 02:00:00.000	2000-10-28 23:00:00.000	180	EDT
EST	2000-10-28 23:00:00.000	2000-10-29 02:00:00.000	120	EST
EST	2000-10-29 02:00:00.000	2001-01-01 00:00:00.000	180	EST

PSTZOFFSET table

The first row of data shows that from January 1, 1999, at midnight, until April 3, 1999, at 11 p.m., (base time—PST, in this case), the offset between EST and the base time zone is 180 minutes, meaning that EST is 180 minutes ahead of PST. The label for EST during this period is *EST*.

The second row shows that from April 3, 1999, at 11 p.m., until April 4, 1999, at 2 a.m., there is a three-hour time period during which EST is 240 minutes ahead of PST. This is because EST has changed to daylight savings time, but PST hasn't changed yet. The label for EST during this period is *EDT*, for eastern daylight time.

Viewing Daylight Savings Start and End Dates

Select PeopleTools, Utilities, International, Time Zones, Daylight Savings to access the Daylight Savings page.

Time Zone Data		Daylight Savings											
Daylight Savings Detail								Customize	Find	View All	First	1-10 of 17	Last
Daylight Savings Time ID	Absolute	Month	Day	Day of the Week	Hour	Minute	Description						
0LastSunMar	<input type="checkbox"/>	Mar	5	Sunday	0	0	Last Sunday in March, midnight	+	-				
1LastSunSep	<input type="checkbox"/>	Sep	5	Sunday	1	0	Last Sunday in Sept, 1:00am	+	-				
22ndSunFeb	<input type="checkbox"/>	Feb	2	Sunday	2	0	Second Sunday in Feb, 2:00am	+	-				
24thTueSep	<input type="checkbox"/>	Sep	4	Tuesday	2	0	Fourth Tuesday in Sept, 2:00am	+	-				
2FirstFriMay	<input type="checkbox"/>	May	1	Thursday	2	0	First Friday in May, 2:00am	+	-				
2FirstSunApr	<input type="checkbox"/>	Apr	1	Sunday	2	0	First Sunday in April, 2:00am	+	-				
2FirstSunMar	<input type="checkbox"/>	Mar	1	Sunday	2	0	First Sunday in March, 2:00am	+	-				
2FirstSunOct	<input type="checkbox"/>	Oct	1	Sunday	2	0	First Sunday in Oct, 2:00am	+	-				
2LastSunMar	<input type="checkbox"/>	Mar	5	Sunday	2	0	Last Sunday in March, 2:00am	+	-				
2LastSunOct	<input type="checkbox"/>	Oct	5	Sunday	2	0	Last Sunday in October, 2:00am	+	-				

Daylight Savings page

Daylight Savings Time ID Displays a unique identifier for the date.

Absolute Clear this check box to indicate a relative date, such as *Last Sunday in October*.

Day For absolute dates, displays the day of the month.

For relative dates, the numbers 1, 2, 3, and 4 represent the first, second, third, and fourth occurrence of the day of the week; for example, to describe the second Sunday in February, enter 2.

The number 5 for a relative date indicates the last occurrence of the specified day of the week, even if that day of the week occurs only four times in the month.

Controlling Time Zone Display When Developing PeopleSoft Application Objects

This section provides an overview of time zone display settings and discusses how to:

- Set time zone options in record field properties.
- Set time zone options in page field properties.
- Use PeopleCode time functions.

Understanding Time Zone Display Settings

Although times are always stored in the base time zone, you can choose whether a page displays a time in the base time or in another time zone.

Time zone display and processing functions operate only on time or datetime fields, not on date fields. This is because all time zone processing requires the knowledge of the time component of the field, which is not provided in date fields. Even if some databases store PeopleSoft date fields internally as datetime fields, the PeopleSoft system does not use the time component of these fields.

Two settings work together to control the time display on pages in the PeopleSoft Pure Internet Architecture:

- Record field properties.

Choose whether the time display is based on the base time zone or on another time zone. If you base the display on another time zone, you can specify a time zone control field, which determines the time zone.

- Page field properties.

Choose whether to automatically include the time zone abbreviation in a field that displays time or datetime values.

Use these guidelines to help decide how to implement time zone displays:

- If the record field properties option specifies that time appears in the base time zone, setting the page field properties to show the time zone helps users interpret the time shown on the page.
- If the record field properties option specifies that the time zone of a particular time or datetime field is determined by the time zone control field, then put that control field directly on the page, to enable the user to manually select the time zone for a field.
- If you put the time zone control field on the page where users are permitted to enter a time zone, disable the page field properties option that includes the time zone abbreviation in the time field.

It's redundant to show the time zone in both places.

- If you do not permit users to enter a time zone manually, for example, if you use the %ClientTimeZone or %BaseTimeZone system variables to set the time zone, then display the time zone either by including the control field on the page or by setting the page field properties appropriately—not both.

In the Microsoft Windows client, users can display time values in their local time zones or in the system's base time zone, regardless of how the record field properties and page field properties are set. By default, all Microsoft Windows client time and datetime fields appear in the base time zone, but each user can override this option by selecting View, Time Display, Local Time or View, Time Display, Database Time.

Setting Time Zone Options in Record Field Properties

When a record includes a time or datetime field, choose whether the PeopleSoft system displays the time in the system's base time zone, or in another time zone such as a time zone that corresponds to the user's location. Set this option for all time and datetime fields.

To set the time zone that controls the display time:

1. In the record, open the Record Field Properties dialog box for the time field by double-clicking the field that displays the time.

The Use tab appears.

2. In the Time Zone group box, set the time zone property.

If you leave the Specified Time Zone check box clear, the time appears in the database's base time zone, unless a user selects View, Time Display in the Microsoft Windows client to display in the user's local time zone.

If you select the Specified Time Zone check box, the system converts the time according to the time zone that is specified in the Time Zone Control Field field. You must specify a time zone control field in this record.

A single time zone control field can control multiple time and datetime fields on the record. Using a single control field causes all times to appear in the same time zone. If you want multiple time or datetime fields to appear in different time zones, you need separate control fields for each of them.

You may want to set the default value of the time zone field to `%ClientTimeZone` or `%BaseTimeZone`. However, the default time zone is used only when the transaction is created. Because the time zone is saved along with the transaction itself, future users who access the transaction will see the time relative to that time zone.

3. Click OK.

Setting Time Zone Options in Page Field Properties

When you place a time or datetime field on a page, you can choose whether to display the time zone in the same field where the time appears.

You would normally want to display the time zone this way when you want users to be aware of the time zone of a field without changing the time zone.

If, on the other hand, you want users to be able to set the time zone, don't display the time zone in the time field. Instead, design your page so that it includes a separate drop-down list box referencing the time zone control field. With this design, users can both see and change the time zone.

To show the time zone in the time field:

1. On the page, open the Page Field Properties dialog box for the time field by double-clicking the page control that displays the time.

The Record tab appears.

2. Select the Display Time Zone box.

When you select this box, the page displays the time zone in the same field as the time and resizes the edit box to allow space for up to 10 additional characters.

This option has no affect on data entry. Users cannot override the time zone. Users who see the time in GMT must enter the time in GMT.

If the field that is associated with this page control is not a time or datetime field, selecting the Display Time Zone box has no effect.

3. Click OK.

Using PeopleCode Time Functions

Two PeopleCode function convert times to their base time zone equivalent: `ConvertTimeToBase` and `ConvertDatetimeToBase`. Another function, `IsDaylightSavings`, establishes whether daylight savings time is in effect on a particular date.

Additionally, you can use system variables `%clienttimezone` and `%basetimezone` in PeopleCode programs or as field default values. The `%clienttimezone` variable returns the current user's local time zone, and `%basetimezone` returns the base time zone of the database.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, "PeopleCode Built-in Functions," Date and Time

CHAPTER 4

Selecting and Configuring Character Sets and Language Input and Output

This chapter provides an overview of character sets and describes how to:

- Select character sets.
- Validate field length.
- Handle international characters in PeopleTools.
- Detect and convert characters.

Understanding Character Sets

This section discusses:

- Character sets.
- Common character sets.
- The Unicode standard.
- Non-Unicode character sets.
- Character sets in the PeopleSoft Pure Internet Architecture.

Character Sets

Before you install your PeopleSoft system you must select an appropriate character set for PeopleSoft client workstations, application servers, and database servers.

A *character set*, also known as a *codepage*, is an ordered set of characters in which each character is mapped to a numeric index, called a *codepoint*. This codepoint stores character data in a computer system. Many hundreds of character sets exist. Some are international standards, sanctioned by the International Organization for Standardization (ISO), some are country-specific standards, and others are not standardized at all, but are specific to a particular computer system vendor. Given the number of separate computers that are involved in a typical PeopleSoft installation, it is likely that your system uses several different character sets.

Common Character Sets

Although there is general agreement on the content and arrangement of most character sets, especially those that are standardized by the ISO, many different names are used by vendors and software packages for similar or identical character sets. US-ASCII encodes the basic characters and symbols that are needed to write the English language. However, US-ASCII is limited to 127 characters and cannot represent many characters that are needed by Western European languages, such as French and German, let alone ideographic languages, such as Japanese and Chinese, in which each character represents a word or concept. Many character sets, however, include all US-ASCII characters in addition to their other characters.

The following table illustrates just a few common character sets that you are likely to encounter and some of the names that are used by different vendors to refer to them:

Character Set	Description and Comments	Type	PeopleSoft and Structured Query Report (SQR) Name	Oracle database management system (DBMS) Name	Microsoft Windows Name	Sybase Name
ISO 8859-1	Western European Latin-1. Contains characters that are required to represent Western European languages. However, does not include the euro symbol, the trademark (TM) symbol, or the <i>oe</i> ligature.	ISO	LATIN1 or ISO_8859-1	WE8ISO8859P1	CP28591	iso_1

Character Set	Description and Comments	Type	PeopleSoft and Structured Query Report (SQR) Name	Oracle database management system (DBMS) Name	Microsoft Windows Name	Sybase Name
Microsoft Codepage 1252	Microsoft Codepage 1252 - Western European. Very similar to ISO 8859-1, except for the inclusion of additional characters. Includes the euro symbol, trademark (TM) symbol, and <i>oe</i> ligature, but using a different codepoint than ISO 8859-15.	Vendor (Microsoft)	CP1252	WE8MSWIN1252	CP1252	cp1252
ISO 8859-2	Central /Eastern European Latin-2. Contains characters that are required for Central European languages, including Czech, Hungarian, and Polish. Does not include the euro symbol.	ISO	LATIN2 or ISO_8859-2	EE8ISO8859P2	CP28592	iso88592

Character Set	Description and Comments	Type	PeopleSoft and Structured Query Report (SQR) Name	Oracle database management system (DBMS) Name	Microsoft Windows Name	Sybase Name
ISO 8859-15	Western European extended Latin-9. Similar to ISO 8859-1, but contains the euro symbol, <i>oe</i> ligature, and several characters that are required for Icelandic.	ISO	LATIN9 or ISO_8859-15	WE8ISO8859P15	CP28605	iso15
Shift-JIS	Most common Japanese character set. Defines thousands of characters for writing Japanese.	Country (Japan)	SJIS	JA16SJIS	CP932	sjis
IBM CCSID 37	IBM Coded Character Set ID 37. Western European Multilingual EBCDIC-based character set.	Vendor (IBM)	EBCDIC	WE8EBCDIC37	CP1140	cp1140

Some of these character sets, such as ISO 8859-1 and IBM CCSID 37, require only one byte to represent each character. For example, in ISO 8859-1, the hexadecimal number 61 represents the lowercase Latin letter *a*. However, larger character sets, such as Shift-JIS, may require more than one byte to represent each character.

The Unicode Standard

The most important consideration when dealing with character sets across a system is ensuring that all characters that you plan to represent within the PeopleSoft system exist in the character set that is used by each component of the system.

For example, if you plan to maintain Japanese characters in employee names, you must ensure that:

- Character sets that are used by the database system include Japanese characters.

- Each external system feeding into or out of the PeopleSoft system expects data in a character set that includes Japanese characters.
- Workstations and printers are installed with fonts that include those characters.

For example, the Japanese Shift-JIS character set contains Japanese and many US-ASCII characters; it is sufficient for encoding both English data and the primary characters that are required in Japanese. However, it does not include the accented Latin characters that are needed for French and German, so it is not a suitable character set for implementations that encompass Western European countries.

Given the sample list of common character sets in the previous table and the number of languages that are required by a typical global PeopleSoft implementation, selecting a character set can be daunting, especially when you are planning to support a large list of languages.

To simplify this type of situation, an industry consortium of vendors devised a universal character set: the Unicode standard. This internationally recognized character set encodes every character that is required to write virtually every written language. The repertoire of characters is jointly maintained by the Unicode Consortium and ISO, and it is synchronized with ISO standard 10646: Universal Multiple-Octet Coded Character Set (UCS).

The PeopleSoft system uses Unicode throughout PeopleTools to simplify character handling. The PeopleSoft system allows the use of Unicode within PeopleSoft databases to enable you to maintain a single database with characters from virtually any language.

Unicode Encodings

Unicode provides space for more than one million characters. To manage such a large repertoire of characters, Unicode defines 16 planes. Each plane contains 65,533 character positions. Plane 0, known as the Basic Multilingual Plane (BMP), is generally sufficient for everyday business use. The other planes are intended to encode extended or rarely used ideographic characters (such as Japanese, Chinese, and Korean ideographs), archaic scripts, and other rarely used characters, such as braille and advanced mathematical symbols.

Note. PeopleTools 8.4 supports the use of characters from the BMP only, otherwise known as plane 0. Support for characters on other planes is planned for a later release of PeopleTools.

Based on the concept of 16 planes, each Unicode character has a unique, 4-byte code. Using 4 bytes of storage per character may seem wasteful, especially considering that most applications today use a small collection of characters and that the vast majority of characters in business applications are from the BMP. Therefore, several different encoding forms for Unicode characters have been standardized and adopted by vendors who are implementing the standard.

Four encodings of Unicode are widely used. All of them are fully compatible with each other and share the same repertoire of characters. They differ, however, in how each character is represented at the byte level. These encodings are:

- Universal Transformation Form - 32-bit (UTF-32)
- Universal Transformation Form - 16-bit (UTF-16)
- Universal Transformation Form - 8-bit (UTF-8)
- Universal Character Set - 2-byte (UCS-2)

PeopleSoft currently supports UCS-2 and UTF-8 encodings; however, following are brief overviews of all four encodings.

Note. The PeopleSoft system supports only UTF-8 and several Asian character sets for outgoing email messages from PeopleTools application servers.

Encoding	Description
UTF-32	<p>The full, 32-bit (4-byte) encoding of Unicode. Each Unicode character is represented by a 4-byte number. For example, the LATIN SMALL LETTER A (a) character is represented in hexadecimal as 0x00000061. UTF-32 was formerly called UCS-4. PeopleSoft systems do not use UTF-32.</p>
UTF-16	<p>An extension of UCS-2 in which the application references characters on planes other than the BMP by combining two UCS-2 code units to designate a single, non-BMP character. UCS-2 is upward compatible with UTF-16 in that each UCS-2 character is also a valid character in UTF-16. However, UTF-16 allows characters outside the BMP to be referenced. These additional characters, known as supplementary characters, require two UTF-16 code units to be represented: a low surrogate and a high surrogate. When no supplementary characters are present, UTF-16 is identical to UCS-2.</p> <p>PeopleSoft plans to use UTF-16 in future versions of PeopleTools, but does not use UTF-16 in PeopleTools 8.45.</p>
UTF-8	<p>A transformation of Unicode that encodes each character as 1 to 4 bytes, depending on which character is being encoded. All US-ASCII characters are encoded in UTF-8 as 1 byte, and this byte is identical to the encoding in US-ASCII. UTF-8 data is therefore backward compatible with US-ASCII data. All characters in the BMP are encoded as 1, 2, or 3 bytes in UTF-8. Characters in other planes are encoded as 4 bytes in UTF-8. UTF-8 has three main advantages: it is fully US-ASCII compatible, US-ASCII data can be considered as UTF-8 data, and it does not require that all characters use 2 or more bytes of storage.</p> <p>PeopleTools uses UTF-8 for serving HTML pages in the PeopleSoft Pure Internet Architecture. PeopleSoft also uses UTF-8 in Oracle and Sybase Unicode databases.</p>
UCS-2	<p>A 16-bit (2-byte) representation of each Unicode character. As such, it can reference only 65,535 codepoints and is limited to characters on the BMP.</p> <p>The PeopleSoft system uses UCS-2 in memory on the Microsoft Windows development tools, the application server, and in Unicode Microsoft SQL Server and IBM DB2 UDB for Linux, Unix, and Windows databases.</p>

The following table summarizes the four Unicode encodings and their uses in the PeopleSoft system:

Unicode Encoding	Minimum Bytes Per Character	Maximum Bytes Per Character	PeopleSoft Usage
UTF-32	4	4	None.
UTF-16	2	4	None.
UTF-8	1	4	PeopleSoft Pure Internet Architecture HTML pages, inbound and outbound XML, and Oracle and Sybase Unicode databases.
UCS-2	2	2	In-memory Microsoft Windows client; application server; Microsoft SQL Server and IBM DB2 UDB for Linux, Unix, and Windows databases Unicode databases.

Encoding Examples

This section includes Unicode encoding examples for the following characters:

a	Latin small letter <i>a</i> .
ñ	Latin small letter <i>n</i> .
か	Hiragana letter <i>ka</i> .

The following table shows examples of these characters in each of the Unicode encodings:

Unicode Encoding	Latin Small Letter <i>a</i>	Latin Small Letter <i>n</i> With Tilde	Hiragana Letter <i>ka</i>
UTF-32	0x00000061	0x000000f1	0x0000304b
UTF-16	0x0061	0x00f1	0x304b
UTF-8	0x61	0xc3b1	0xe3818b
UCS-2	0x0061	0x00f1	0x304b

See Also

The Unicode Standard Version 3.0, The Unicode Consortium, Reading, MA, USA, Addison-Wesley Developers Press, 2000. ISBN 0-201-61633-5

<http://www.unicode.org/unicode/uni2book/u2.html>

[Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,”
Selecting Email Character Sets, page 54](#)

Non-Unicode Character Sets

Although much of the PeopleSoft system runs by using Unicode, you can configure several components with a non-Unicode character set. When making these choices, you should understand the types of character sets other than Unicode that exist.

This section discusses:

- SBCSs.
- DBCSs.

Note. For the sake of terminology, some systems, such as Microsoft Windows, refer to two types of character sets: Unicode and ANSI. ANSI, in this context, refers to the American National Standards Institute, which maintains equivalent standards for many national and international standard character sets. In the context of this book, ANSI character sets refer to non-Unicode character sets, which can be any international, national, or vendor standard character set, such as those that are discussed at the beginning of this chapter.

SBCSs

Most character sets use 1 byte to represent each character and are therefore known as SBCSs. These character sets are relatively simple and can represent up to 255 unique characters. Examples of SBCSs are ISO 8859-1 (Latin1), ISO 8859-2 (Latin2), Microsoft CP1252 (similar to Latin1, but vendor specific), and IBM CCSID 37.

DBCSs

DBCSs use 1 or 2 bytes to represent each character and are typically used for writing ideographic scripts, such as Japanese, Chinese, and Korean. Most DBCSs allow a mix of 1-byte and 2-byte characters, so you cannot assume an even-string byte length.

The PeopleSoft system supports two types of DBCS:

- Nonshifting
- Shifting

The difference between these types of DBCS is in the way in which the system determines whether a particular byte represents 1 character or is part of a 2-byte character.

Nonshifting DBCSs

Nonshifting DBCSs use ranges of codepoints, specified by the character set definition, to determine whether a particular byte represents 1 character or is part of a 2-byte character.

In nonshifting DBCSs, the 2 bytes that are used to form a character are called *lead bytes* and *trail bytes*. The lead byte is the first in a 2-byte character, and the trail byte is the last. Nonshifting DBCSs differentiate single-byte characters from double-byte characters by the numerical value of the lead byte. For example, in the Japanese Shift-JIS encoding, if a byte

is in the range 0x81-0x9F or 0xE0-0xFC, then it is a lead byte and must be paired with the following byte to form a complete character.

The most popular client-side Japanese codepage, Shift-JIS, uses this lead byte/trail byte encoding scheme, as do most Microsoft Windows and UNIX/Linux ASCII-based double-byte character sets that represent Chinese, Japanese, and Korean characters. Contrary to its name, Shift-JIS is a nonshifting double-byte character set.

Shifting DBCSs

A shifting DBCS is another double-byte encoding scheme in use that doesn't use the lead byte and trail byte concept. The IBM DB2 UDB for OS/390 and z/OS EBCDIC-based Japanese, Chinese, and Korean character sets use this shifting encoding scheme.

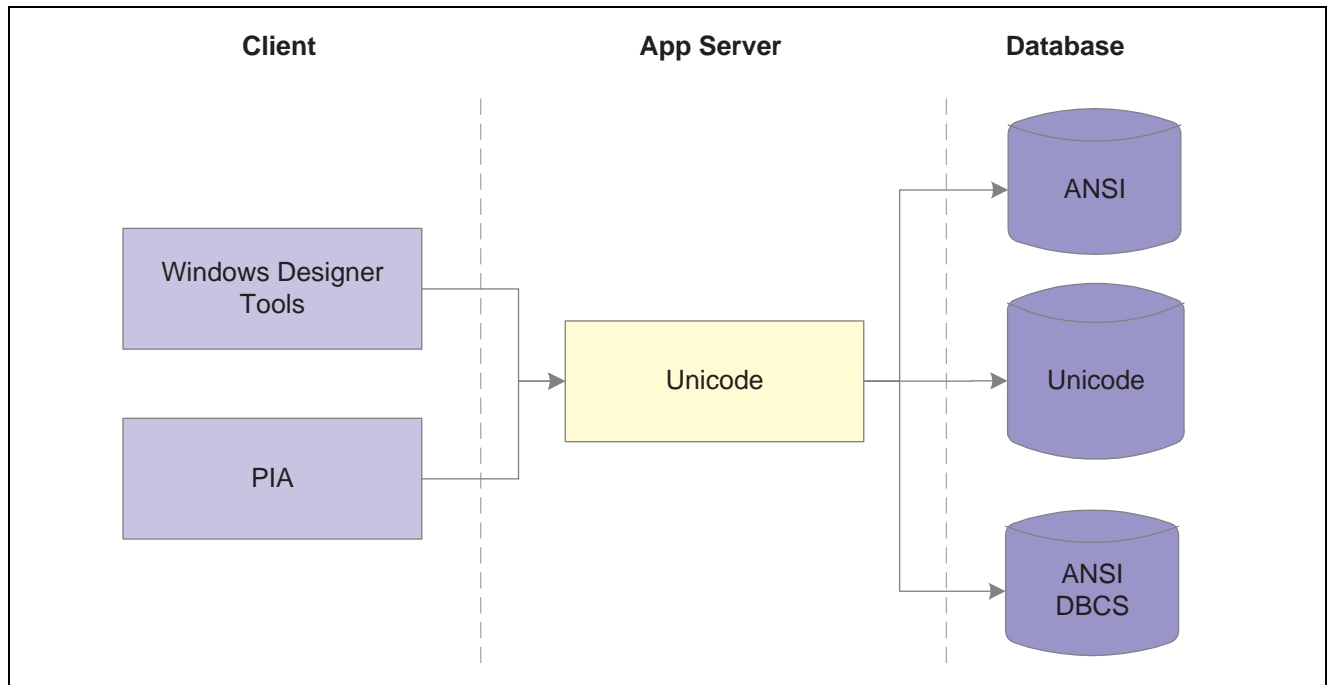
Instead of reserving a range of bytes as lead bytes, shifting DBCSs always keep track of whether they are in double-byte or single-byte mode. In double-byte mode, every two bytes form a character. In single-byte mode, every byte is a character in itself. To track what mode the character set is in, the system uses *shifting* characters. By default, the character set is expecting single-byte data. As soon as a double-byte character needs to be represented, a shift-in byte is added to the string. From this point on, all characters are expected to be 2 bytes per character. This continues until a shift-out byte is detected, which indicates that the character set should go back to a single byte per character.

This scheme, while more complex than the lead byte and trail byte scheme, provides greater performance, because the system always knows how many bytes should be in any particular character. Unfortunately, it also increases the length of the string. For example, a character string that comprises a mixture of single-byte and double-byte characters requires more space to store in a shifting character set because you need to include the shift-in and shift-out bytes, as well as the data itself.

Character Sets in the PeopleSoft Pure Internet Architecture

PeopleSoft installations include multiple components, each of which must handle differing character sets.

This figure illustrates the PeopleSoft Unicode architecture:



PeopleSoft Unicode architecture

PeopleSoft clients and application servers use Unicode exclusively and do not rely on other character sets to represent and process data. However, depending on your environment, not all system components support Unicode-encoded data, so you may not be able to run all parts of your system in Unicode. Therefore, PeopleTools enables you to configure these system components to use other character sets:

- The PeopleSoft database.

Not all database platforms support Unicode data storage. Even when Unicode storage is available, not all PeopleSoft implementations require the flexibility in language choices that are available with Unicode, so you can use a non-Unicode encoding for the database.

- COBOL.

The character set that is used for PeopleSoft COBOL processing must match the character set of the database. If you created a Unicode database for the PeopleSoft implementation, you must also run COBOL in Unicode.

- File input and output.

All file operations in PeopleTools, including file layout objects, trace and log files, and file operations from SQRs can be performed in Unicode or any supported non-Unicode character set. This is useful in situations in which you must interface with an external system that does not support Unicode.

- Third-party products that are non-Unicode compliant.

Some third-party products that are supported by PeopleTools do not yet support Unicode. In this case, PeopleTools converts application data to a specific, non-Unicode character set before communicating with these tools. Some of the products that do not yet support Unicode data are Hyperion Essbase and Cognos PowerPlay.

When Unicode is not used for any of these types of operations or data storage, the PeopleSoft system transparently handles the conversion from Unicode to a non-Unicode character set. The non-Unicode character set that is used depends on several settings, which are discussed in detail later in this chapter.

The following table lists the character sets that the PeopleSoft system supports and the names by which they may be referred to in PeopleSoft applications. Use these character set names:

- In PeopleCode programs for manipulating file layout objects.
- In the UNIX/Linux application server configuration to determine the default, non-Unicode character set for log files, trace files, and operating system interfaces.
- When creating your database.

A limited number of the character sets in this table are supported as database character sets. Refer to your hardware and software requirements guide for details about the character sets that are supported for your database platform.

This PeopleBook also contains information about supported character set encodings for SQR for PeopleSoft globalization.

See [Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” SQR for PeopleSoft-Supported Character Set Encodings, page 124.](#)

See *PeopleTools 8.45 Hardware and Software Requirements Guide*

Character Set Name	Description and Comments	Character Set Type
ANSI	Current ANSI-based code page. Not really a character set, but causes the system to use the default non-Unicode character set of the host operating system. With Microsoft Windows NT, the ANSI codepage is dependent on the language version of the Windows operating system. For example, Japanese Windows NT uses CP932 as its ANSI codepage. English Windows NT uses CP1252 as its ANSI codepage. Windows 2000 and XP enable the system administrator to select the ANSI codepage independently of the language release of Windows.	SBCS or DBCS, depending on the host operating system.
ASCII	7-bit US-ASCII	SBCS
Big5	Big5 (Traditional Chinese)	Nonshifting DBCS
CCSID1027	IBM EBCDIC 1027 (Japanese-Latin)	SBCS
CCSID1047	IBM EBCDIC 1047 (Latin1)	SBCS
CCSID290	IBM EBCDIC 290 (Katakana)	SBCS
CCSID300	IBM EBCDIC 300 (Kanji)	Nonshifting DBCS

Character Set Name	Description and Comments	Character Set Type
CCSID930	IBM EBCDIC 930 (Kana-Kanji)	Shifting DBCS
CCSID935	IBM EBCDIC 935 (Simplified Chinese)	Shifting DBCS
CCSID937	IBM EBCDIC 937 (Traditional Chinese)	Shifting DBCS
CCSID939	IBM EBCDIC 939 (Latin-Kanji)	Shifting DBCS
CCSID942	IBM EBCDIC 942 (Japanese PC)	Nonshifting DBCS
CP1026	Windows 1026 (EBCDIC)	SBCS
CP1250	Windows 1250 (Eastern Europe)	SBCS
CP1251	Windows 1251 (Cyrillic)	SBCS
CP1252	Windows 1252 (Latin1)	SBCS
CP1253	Windows 1253 (Greek)	SBCS
CP1254	Windows 1254 (Turkish)	SBCS
CP1255	Windows 1255 (Hebrew)	SBCS
CP1256	Windows 1256 (Arabic)	SBCS
CP1257	Windows 1257 (Baltic)	SBCS
CP1258	Windows 1258 (Vietnamese)	SBCS
CP1361	Windows 1361 (Korean Johab)	SBCS
CP437	MS-DOS 437 (US)	SBCS
CP500	Windows 500 (EBCDIC 500V1)	SBCS
CP708	Windows 708 (Arabic - ASMO708)	SBCS

Character Set Name	Description and Comments	Character Set Type
CP720	Windows 720 (Arabic - ASMO)	SBCS
CP737	Windows 737 (Greek - 437G)	SBCS
CP775	Windows 775 (Baltic)	SBCS
CP850	MS-DOS 850 (Latin1)	SBCS
CP852	MS-DOS 852 (Latin2)	SBCS
CP855	MS-DOS 855 (IBM Cyrillic)	SBCS
CP857	MS-DOS 857 (IBM Turkish)	SBCS
CP860	MS-DOS 860 (IBM Portuguese)	SBCS
CP861	MS-DOS 861 (Icelandic)	SBCS
CP862	MS-DOS 862 (Hebrew)	SBCS
CP863	MS-DOS 863 (Canadian French)	SBCS
CP864	MS-DOS 864 (Arabic)	SBCS
CP865	MS-DOS 865 (Nordic)	SBCS
CP866	MS-DOS 866 (Russian)	SBCS
CP869	MS-DOS 869 (Modern Greek)	SBCS
CP870	Windows 870	SBCS
CP874	Windows 874 (Thai)	SBCS
CP875	Windows 875 (EBCDIC)	SBCS
CP932	Windows 932 (Japanese)	Nonshifting DBCS

Character Set Name	Description and Comments	Character Set Type
CP936	Windows 936 (Simplified Chinese)	Nonshifting DBCS
CP949	Windows 949 (Korean)	Nonshifting DBCS
CP950	Windows 950 (Traditional Chinese)	Nonshifting DBCS
EBCDIC	IBM EBCDIC CCSID37 (USA)	SBCS
EUC-JP	Extended UNIX code (Japanese)	Nonshifting DBCS
EUC-KR	Extended UNIX code (Korean)	Nonshifting DBCS
EUC-TW	Extended UNIX code (Taiwan)	Nonshifting DBCS
EUC-TW-1986	Extended UNIX code (TW-1986)	Nonshifting DBCS
GB12345	GB 2312 (Simplified Chinese)	Nonshifting DBCS
GB2312	GB 2312 (Simplified Chinese)	Nonshifting DBCS
ISO-2022-JP	ISO-2022-JP Japanese	Shifting DBCS
ISO-2022-KR	ISO-2022-JP Korean	Shifting DBCS
ISO_8859-1	ISO 8859-1 (Latin1)	SBCS
ISO_8859-10	ISO 8859-10 (Latin6)	SBCS
ISO_8859-11	ISO 8859-11 (Thai)	SBCS
ISO_8859-14	ISO 8859-14 (Latin8)	SBCS
ISO_8859-15	ISO 8859-15 (Latin9/Latin0)	SBCS
ISO_8859-2	ISO 8859-2 (Latin2)	SBCS
ISO_8859-3	ISO 8859-3 (Latin3)	SBCS

Character Set Name	Description and Comments	Character Set Type
ISO_8859-4	ISO 8859-4 (Latin4)	SBCS
ISO_8859-5	ISO 8859-5 (Cyrillic)	SBCS
ISO_8859-6	ISO 8859-6 (Arabic)	SBCS
ISO_8859-7	ISO 8859-7 (Greek)	SBCS
ISO_8859-8	ISO 8859-8 (Hebrew)	SBCS
ISO_8859-9	ISO 8859-9 (Latin5)	SBCS
JIS_X0201	Japanese Half-width Katakana	Nonshifting DBCS
JIS_X_0208	Japanese Kanji	Nonshifting DBCS
Java	Java (Unicode encoding)	Unicode
Johab	Johab (Korean)	Nonshifting DBCS
Shift_JIS	Shift-JIS (Japanese)	Nonshifting DBCS
UCS2	Unicode UCS-2	Unicode
UTF7	Unicode UTF-7. (An outdated Unicode 7-bit clean transformation that is sometimes used for email that must pass through gateways that do not support 8-bit characters.)	Unicode
UTF8	Unicode UTF-8	Unicode

Note. Although the PeopleSoft system's use of Unicode incorporates data in all languages, it does not currently support a user interface in Arabic or Hebrew. These languages require bidirectional rendering because they are written with a right-to-left general orientation.

See Also

For more information and code charts for Microsoft codepages, visit <http://www.microsoft.com/globaldev/reference/cphome.asp>

For more information and code charts for the ISO 8859 series of character sets, visit <http://czyborra.com/charsets/iso8859.html>

For more information and code charts for Unicode, visit <http://www.unicode.org>

Selecting Character Sets

This section discusses how to:

- Select database character sets.
- Select application server character sets.
- Select and manage client workstation character sets.
- Select email character sets.

See Also

[Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” Understanding Character Sets, page 35](#)

Selecting Database Character Sets

The primary character set decision that you must make when installing a PeopleSoft implementation is which character set to use for the database system. Ideally, all databases are encoded in Unicode; however, in some cases Unicode requires several bytes to represent each character when only 1 byte may be required in a non-Unicode character set. Therefore, the PeopleSoft system enables you to use certain non-Unicode character sets for the database.

By using a Unicode encoded database, you can maintain a single database with data in any combination of languages. A single PeopleSoft application server can serve multiple users connecting to the mixed-language database, regardless of the language or character set of those users' client machines. The only restriction on a user's ability to access mixed-language data is the capability of the user's client workstation to interpret, display, and accept keyboard entry of the characters from the various languages.

Most language or region-specific non-Unicode character sets provide sufficient characters for only a few languages. If you create a non-Unicode database, you must ensure that all of the characters for all of the languages that you plan on using can be represented in the character set that you choose.

Depending on the data that you store and how the database stores Unicode characters, a Unicode database can be significantly larger than a non-Unicode database. However, only the storage of character data is affected; the space that is required for non-character data, such as numbers and dates (which are stored by the database system as numbers), is not affected.

Depending on the database platform, you can use one of the four character set types (SBCS, nonshifting DBCS, shifting DBCS, or Unicode) when creating the database. However, the number of characters that you can store in each column is affected greatly by the type of character set that you choose for the database encoding.

See Also

PeopleTools 8.45 Hardware and Software Requirements Guide

PeopleTools 8.45 Installation Guide for Your Database Platform

Your operating system and database guides

Selecting Application Server Character Sets

All data that is stored in memory and processed by the PeopleTools application server is held in Unicode. However, the application server allows files on the server (created through PeopleCode file layout objects) and log and trace files to be Unicode or non-Unicode. Although the PeopleSoft application server uses Unicode internally for all data processing, it can create these files in Unicode or in a non-Unicode character set.

Each PeopleSoft application server is configured with a default non-Unicode character set. If a file operation must create a non-Unicode file, this character set is used, unless another character set is explicitly specified in the file operation. For example, if you create a file layout object to write a non-Unicode file, but you don't specify in which character set the file should be created, the default non-Unicode character set of the application server is used.

On Microsoft Windows application servers, the operating system's default system character set is used for the PeopleSoft application server's default non-Unicode character set. In Windows NT 4.0, this default character set is hard-wired into the operating system, based on the language of the Windows version being used, and you cannot change it. For example, in English Windows NT 4.0, the default non-Unicode character set is Microsoft CP1252. In Japanese Windows NT 4.0, the default non-Unicode character set is Microsoft CP932. You can determine the default character set of your Windows machine by issuing the `chcp` command at a command prompt.

Microsoft Windows 2000 and XP enable you to change the default character set of the system, although as installed, the default character set matches the default locale of the Windows 2000 or XP installation. To change the system default locale (and therefore the character set), on Windows 2000 servers, use the Control Panel's Regional Options applet. In the Language settings for the system section, click the Set Default button.

When running on UNIX/Linux, the PeopleSoft application server enables you to specify the default non-Unicode character set in the application server's configuration file, which you select by using the PSADMIN tool. Any valid PeopleSoft character set with a character set type of SBCS or nonshifting DBCS is a valid default non-Unicode character set for PeopleSoft application servers that run on UNIX/Linux.

See Also

[Chapter 4, "Selecting and Configuring Character Sets and Language Input and Output," Character Sets in the PeopleSoft Pure Internet Architecture, page 43](#)

Selecting and Managing Client Workstation Character Sets

You must consider the client components of PeopleTools when you are planning your language strategy. The requirements for language support on client workstations are different, depending on whether you are using the PeopleSoft Pure Internet Architecture or the PeopleTools development tools for Microsoft Windows.

This section discusses:

- Character sets and fonts in the PeopleSoft Pure Internet Architecture.
- Fonts and the Microsoft Windows client.
- Input methods.

Character Sets and Fonts in the PeopleSoft Pure Internet Architecture

The PeopleSoft Pure Internet Architecture serves all HTML pages in the UTF-8 encoding of Unicode. This encoding is recognized automatically by the web browser, because the encoding of the page is announced in the HTTP header when the browser communicates with the web server. Most modern browsers can support the use of UTF-8 encoded HTML pages, including all browsers that are supported by PeopleTools.

However, the browser needs other components to correctly display and enter the vast array of characters that are available in Unicode. Specifically, you need appropriate fonts to display the various scripts in which you expect data to be maintained, and you may need alternative keyboard layouts or, in the case of ideographic scripts such as Chinese, Japanese, and Korean, you need input method editors (IMEs) to convert sequences of keystrokes into ideographs. The requirement for alternate keyboard and IMEs is the same for both the PeopleSoft Pure Internet Architecture and the Microsoft Windows client.

Not all fonts contain a full repertoire of Unicode characters, because many fonts are tailored to address a specific list of languages and contain only the glyphs that are required by those languages. If you try to view Unicode data with a font that does not contain the appropriate characters for the displayed language, you will most likely see square boxes in place of the appropriate characters. The data has not been corrupted; there is just no glyph available in the current font for the character that the system is trying to display. For this reason, you may need to license or configure several fonts for a global PeopleSoft system.

The PeopleSoft Pure Internet Architecture includes a set of style sheets, defined with PeopleSoft Application Designer, that determine the font that is used to display HTML pages. In some cases, the application data may contain characters that are not present in this font and that require a different font. For example, none of the default fonts in the English version of Microsoft Windows NT 4.0 include sufficient Japanese characters to display most Japanese translated data.

You may need to obtain and configure fonts that contain the characters for the languages that you are planning to use, if your workstations are not already configured with these fonts. Obtain fonts from the following sources:

- Many Microsoft Windows and other operating system applications are packaged with Unicode fonts containing glyphs covering a large range of languages.

Microsoft Office 2000 is packaged with several fonts containing a large portion of the characters in Unicode, including the Microsoft Sans Serif font. Use these fonts in the PeopleSoft Pure Internet Architecture by specifying them in the PeopleSoft Application Designer style sheet definitions or by following the browser-specific instructions in this section.

- Several public domain fonts exist that contain a large character repertoire for use in web browsers. For a list of such fonts, see <http://www.hclrss.demon.co.uk/unicode/fonts.html>.
- Several font foundries license fonts for individual or corporate use.

Some of these foundries include Agfa Monotype, Bitstream, and Tiro Typeworks.

Depending on your browser, you can also download fonts from your browser's manufacturer.

Microsoft Internet Explorer 5.0 or later enables you to download fonts for most popular languages from the Microsoft web site. Trigger these downloads by visiting the Internet Explorer website at <http://www.microsoft.com/ie> and downloading the Internet Explorer Language pack in the language of your choice. Alternatively, select the encoding in Internet Explorer for a language for which you want to download fonts. This causes Internet Explorer 5.0 (or later) to automatically download the font pack from the Microsoft website. After this download is completed, reset your encoding to Auto-Select so that PeopleSoft Pure Internet Architecture pages appear correctly.

Microsoft Internet Explorer automatically selects the appropriate font for displaying data in the language that is presented in the HTML page. The appropriate font is the one that most closely matches the font that is specified by the PeopleSoft system in the page's style sheet definition.

Netscape Navigator 6.0 also performs this automatic font matching, based on the language of the page, but does not provide international fonts.

Netscape Navigator 4.0 versions do not perform automatic font matching, and you must configure the browser manually if the font that is specified on the style sheet does not exist on particular workstations. All PeopleSoft Pure Internet Architecture pages are served in the UTF-8 Unicode encoding, so you must change the font setting for the Unicode character set.

Note. The PeopleSoft system does not support the display or maintenance of Japanese, Chinese, and Korean data by using Netscape 4 due to significant limitations in that browser's handling of UTF-8 encoded data. To view or maintain Japanese, Chinese, and Korean data by using the PeopleSoft Pure Internet Architecture, you must use Microsoft Internet Explorer 5.0 or later or Netscape Navigator 6 or later.

See PeopleSoft Platforms Database on Customer Connection

Fonts and the Microsoft Windows Client

The Microsoft Windows client enables you to specify the font that is used for all graphical components in PeopleSoft Application Designer, the Windows client panel processor, and all other PeopleTools modules for Windows. Three separate areas enable you to specify a font:

- Configuration Manager font setting (Display tab)

This setting affects the font that is used by all of the designer components of PeopleTools, including all of the text that is contained in the Microsoft Windows resource files

See [Chapter 14, "Translating PeopleTools," page 209](#).

Changing this font setting may be necessary if your workstation's default locale does not contain the characters that are used for the language that you are attempting to display or maintain. For example, if you are attempting to view Japanese characters on an English Windows workstation, you can change the PeopleSoft Configuration Manager font setting to select a font that contains the characters for the language that you are trying to display.

Several fonts that are shipped with Microsoft Windows 2000 and Microsoft Office 2000, including Arial Unicode MS and Microsoft Sans Serif, contain a large number of glyphs covering most of the languages that are supported by Unicode. Microsoft Windows 2000 and Windows XP can also be configured with fonts for most worldwide languages by selecting the required languages under the Regional Settings Control Panel applet.

- PeopleCode font

The PeopleCode editor in PeopleSoft Application Designer also enables you to select a font for character display in the editor's window itself. This is useful if the PeopleCode programs that you are working on contain Unicode characters. To set the font in PeopleSoft Application Designer, select Tools, Options and then select the PeopleCode tab.

Input Methods

If users will enter foreign language data by using PeopleSoft Pure Internet Architecture or the Microsoft Windows client, you must ensure that an appropriate keyboard layout or input method editor is installed on the workstation.

Most alphabetic languages can be typed by using a relatively simple keyboard layout. Several specialized keyboard layouts exist for most languages; configure these keyboard layouts through your operating system. For example, a Spanish keyboard layout contains keys for the n-tilde character (ñ) and several other accented characters.

There are several ways of entering these characters by using a nonlocalized keyboard. Your operating system manual can help you use specialized keyboard layouts, such as the English International layout, which enables you to enter accented characters by using two keystrokes. The Microsoft web site contains information about keyboards that are supported by Microsoft Windows and instructions for installing and configuring Windows keyboard layouts.

Ideographic languages, such as Chinese, Japanese, and Korean, require the use of a front-end processor to intercept multiple keyboard strokes and transform them into an ideographic character. These are known as IMEs, and they must be installed on each workstation where you plan to enter the ideographic languages.

Most localized versions of operating systems for these languages come preconfigured with IMEs that are appropriate for the language that is supported by the operating system. But on systems where the default locale is not Chinese, Japanese, or Korean, you may need to configure or license an IME from a third-party vendor. The PeopleSoft Pure Internet Architecture supports any IME that is supported by your browser. The designer tools in Microsoft Windows support all standard Microsoft IMEs.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Application Designer, “Creating Style Sheet Definitions”

Selecting Email Character Sets

The PeopleSoft system supports UTF-8 for outgoing Simple Mail Transfer Protocol (SMTP) email messages from PeopleTools application servers. In addition, because many email products in Asian markets support only traditional character sets, the PeopleSoft system supports several additional character sets for outgoing email.

PeopleSoft application servers support the following character sets for outgoing email:

- UTF-8 (default).
- ISO-2022-JP, Shift_JIS, EUC-JP (for Japanese).
- ISO-2022-KR, EUC-KR (for Korean).
- GBK, Big5 (for Chinese).

Specifying Email Character Sets

You specify an email character set in the SMTPCharacterSet parameter in the application server configuration file, psappsrv.cfg. By default, the SMTPCharacterSet parameter is set to UTF-8.

Note. You should specify a value for the SMTPCharacterSet. If you do not specify a value for the parameter, email is sent as-is, with no encoding. Leave the parameter set to the default value of UTF-8 if you are not certain about which value to use.

For example, to use ISO-2022-JP encoding for outgoing SMTP mail, in the psappsrv.cfg file, set the SMTPCharacterSet parameter to ISO-2022-JP, as shown in the following example:

```
[SMTP Setting]
..
SMTPCharacterSet=ISO-2022-JP
SMTPEncodingDLL=<blank>
```

SMTP settings

Writing Custom SMTP Encoding Modules

You can also write your own SMTPEncodingDLL modules, if necessary.

Validating Field Length

This section provides overviews PeopleSoft Application Designer field length semantics and field length checking for non-Unicode databases and discusses how to enable or disable data field length checking.

Understanding Application Designer Field Length Semantics

The database character set determines the way that PeopleTools interprets the column length that is defined in PeopleSoft Application Designer.

If you create a Unicode database, the field length, as shown in PeopleSoft Application Designer, indicates the maximum number of Unicode BMP characters that are permitted in the field, regardless of the Unicode encoding that is used by the database. Some database platforms, such as Oracle, use byte lengths to measure column sizes when operating in a Unicode database, while others use character lengths.

When the database uses byte-sized column lengths, the PeopleSoft system sizes the database columns based on the worst-case ratio between bytes and characters in the Unicode encoding that is used by your database. For example, the UTF-8 character set is used by Oracle; therefore, the worst-case character-to-byte ratio, when running against an Oracle Unicode database, is 1:3, so column size is tripled when creating a Unicode database on Oracle. A field that is defined in PeopleSoft Application Designer as a CHAR(10) is created on an Oracle Unicode database with a type of VARCHAR2(30). This tripling of the maximum column size does not affect the actual size of the database, because variable length character fields do not reserve space in the database.

Other database platforms use character-based column lengths whose sizes represent the maximum number of Unicode characters instead of bytes that may be stored. Examples of this implementation are the NCHAR data type in Microsoft SQL Server and the GRAPHIC data type in DB2 UDB for Linux, Unix, and Windows.

If you create a non-Unicode database, the field length in PeopleSoft Application Designer represents the number of bytes that are permitted in the field, based on the character set that you used to create the database. Therefore, a PeopleSoft Unicode database enables you significantly more space for character data within the database when dealing with ideographic languages, such as Japanese, that require more than 1 byte storage per character.

The following tables show some of the possible database encodings for database platforms that the PeopleSoft system supports in Unicode and DBCS and their effects on database column sizes.

For Oracle:

Database Character Set	Database Representation of a Character Field with a Length of 10 in PeopleSoft Application Designer	Number of Characters (Worst Case) Allowed in a Character Field With a Length of 10 in PeopleSoft Application Designer
Unicode (UTF-8)	VARCHAR2(30)	10
Any SBCS	VARCHAR2(10)	10
Shift-JIS (JA16SJIS)	VARCHAR2(10)	5

For Microsoft SQL Server:

Database Character Set	Database Representation of a Character Field with a Length of 10 in PeopleSoft Application Designer	Number of Characters (Worst Case) Allowed in a Character Field With a Length of 10 in PeopleSoft Application Designer
Unicode (UCS-2)	NCHAR(10)	10
Any SBCS	CHAR(10)	10
Shift-JIS (CP932)	CHAR(10)	5

For Sybase:

Database Character Set	Database Representation of a Character Field with a Length of 10 in PeopleSoft Application Designer	Number of Characters (Worst Case) Allowed in a Character Field With a Length of 10 in PeopleSoft Application Designer
Unicode (UTF-16)	UNIVARCHAR(10)	10
Any SBCS	CHAR(10)	10
Nonshifting DBCS	CHAR(10)	5

For DB2 UDB for OS/390 and z/OS:

Database Character Set	Database Representation of a Character Field with a Length of 10 in PeopleSoft Application Designer	Number of Characters (Worst Case) Allowed in a Character Field With a Length of 10 in PeopleSoft Application Designer
Any SBCS	CHAR(10)	10
Shifting DBCS (CCSID 930/939)	CHAR(10)	4 (4 x 2 byte characters, plus shift-in & shift-out bytes)

For all others:

Database Character Set	Database Representation of a Character Field with a Length of 10 in PeopleSoft Application Designer	Number of Characters (Worst Case) Allowed in a Character Field With a Length of 10 in PeopleSoft Application Designer
Any SBCS	CHAR(10)	10

Understanding Field Length Checking for Non-Unicode Databases

The maximum number of characters that are permitted in a PeopleSoft field varies, depending on the character set of the database. Because all components of PeopleTools use Unicode for internal storage, by default, field length checking occurs in terms of Unicode character counts. This calculation is appropriate for Unicode databases and for any SBCS databases.

However, if you are using a non-Unicode DBCS, special length checking must occur each time you move off a field to ensure that the string that you entered fits in the database column when the string is converted to the database's character set.

For graphically sizing page fields, PeopleTools uses the Unicode length of the field as defined in PeopleSoft Application Designer. For example, if a field is defined in PeopleSoft Application Designer as a 10-character field, page fields in both the PeopleSoft Pure Internet Architecture and the PeopleTools clients for Microsoft Windows allow 10 characters to be displayed unless manually resized by the developer.

However, if the database is encoded in a non-Unicode DBCS character set, such as Japanese Shift-JIS or IBM CCSID 930/939, special length validation must occur because the database column size is created relative to a byte count, not to a character count as is used by the simple field length validation.

For example, if a user enters 10 Japanese characters into a field that is defined as CHAR(10) in PeopleSoft Application Designer, this string needs 20 bytes of storage in a nonshifting DBCS character set and 22 bytes of storage in a shifting character set. This 10-character input would fail insertion in both of these databases.

To address this issue, the page processor checks the Data Field Length Checking option on the PeopleTools Options page and performs character-set specific length validation against the contents of each field when the field is validated. Typically length validation occurs when the field's FieldChange PeopleCode event fires, so the actual time of validation may differ, depending on whether your page uses deferred mode processing.

Enabling or Disabling Data Field Length Checking

To enable or disable data field length checking:

1. Select PeopleTools, Utilities, Administration, PeopleTools Options.

The PeopleTools Options page appears.

2. From the Data Field Length Checking drop-down list box, select a value based on the character set that you are using for the database:

Others

Select if you are using a Unicode encoded database or a non-Unicode SBCS database. This option prevents special field length checking, which is not required by these types of databases.

DB2 MBCS

Select if you are running a Japanese database on the DB2 UDB for Linux, Unix, and Windows platform. This options enables field length checking based on a shifting DBCS.

MBCS Select if you are running a non-Unicode Japanese database on any other platform. This option enables field length checking based on a nonshifting DBCS.

Note. The non-Unicode DBCS settings are specifically oriented to Japanese language installations, because Japanese is the only language that the PeopleSoft system supports in a non-Unicode DBCS encoding. All languages other than Western European languages and Japanese are supported by the PeopleSoft system only when using Unicode encoded databases.

3. Save the page.

Handling International Characters in PeopleTools

PeopleTools contains several features that enable you to manipulate character data based on the language, script, or type of character being processed. Some field formats, such as the Name format, are dependent on the language of the data being maintained. Several PeopleCode functions enable you to manipulate language-dependent characters.

This section discusses standard name formats for Chinese, Japanese, and Korean characters.

Standard Name Formats for Chinese, Japanese, and Korean Characters

The PeopleSoft standard name format is:

```
[lastname] [suffix],[prefix] [firstname] [middle name/initial]
```

However, if the name contains any of the following types of characters, you must separate the first and last names by a space, instead of a comma:

- Chinese, Japanese, and Korean Unified Ideographs (Chinese Hanzi, Japanese Kanji, and Korean Hanja).
- Japanese half-width or full-width Katakana.
- Japanese Hiragana.
- Korean Hangul.

The following example shows the PeopleSoft standard name format when any one of the previous types of characters appear:

```
[lastname] [suffix] [prefix] [firstname] [middle name/initial]
```

This format allows for Chinese, Japanese, and Korean names, which are traditionally separated by spaces, and which would appear incorrectly if separated by a comma.

Detecting and Converting Characters

PeopleTools also provides PeopleCode string functions that recognize and convert between different characters within the Japanese character set. This enables you to detect, convert, and enforce the types of characters that you can enter in any PeopleSoft field. The PeopleSoft system uses these functions in the development of Alternate Character Architecture in some PeopleSoft applications. Alternate Character Architecture is used in several PeopleSoft applications to provide a feature that enables the entry of, and enforces the characters contained in, Japanese phonetic spellings (Furigana) by using the Hiragana or Katakana scripts.

The following PeopleCode string functions recognize and convert between different characters within the Japanese character set:

- CharType
- ContainsCharType
- ContainsOnlyCharType
- ConvertChar

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “PeopleCode Built-in Functions,” CharType

CHAPTER 5

Controlling Currency Display Format

This chapter provides an overview of currency-specific settings and discusses how to:

- Set up currency amount fields when developing applications.
- Set currency field display properties when developing applications.
- Use system-wide multicurrency settings.
- Resize currency fields by using the international field size utility.

Understanding Currency-Specific Settings

PeopleTools can format a currency amount that appears on a page or a report based on both your numerical format preferences and the currency that the amount represents.

Some display settings, such as the characters that are used for the thousands and decimal separators, are based on your language preference. Others, such as the currency symbol and the decimal precision, are based on the properties of the currency in which the amount appears. Those properties include:

- A currency symbol, such as \$ for the Australian dollar.
- A decimal precision.

For example, there are two decimal positions for Australian dollars (\$5.00), but no decimal positions for the Korean Won (500 KRW).

Currency information is stored in the Currency Code table (CURRENCY_CD_TBL) record definition along with other information about the currencies that are used in your implementation. The currency symbol is controlled by the CUR_SYMBOL field. The decimal precision for a currency is controlled by the DECIMAL_POSITIONS field.

Note. PeopleSoft provides updates to the Currency Code table containing the latest values from the ISO Standard 4217, “Codes for the representation of currencies,” with each PeopleTools minor release. Even though PeopleSoft provides these standardized values, a full list of the symbols that are used by each currency is not provided, as this is not part of the ISO standard. Use the Currency Code page to check and set the appropriate currency symbol for each currency that you plan to use in your implementation.

The navigation to the page where you maintain currency code information depends on which applications you are using.

See *PeopleTools 8.45 Installation Guide for Your Database*

Setting Up Currency Amount Fields When Developing Applications

To set up currency amount fields:

1. Access PeopleSoft Application Designer.
2. Ensure that the record definition has an appropriate currency control field.

The currency control field must be in the same record as the currency display field and hold a valid Currency Code value. This means that it must be a character (Char) field with a length of 3, formatted in uppercase (Upper). You can use an existing field or define a new field. The field does not need to be named CURRENCY_CD. It can have any valid PeopleSoft field name. However if the record has only one amount field, you should use the CURRENCY_CD field name for consistency with other PeopleSoft applications.

An example of a record with two currency-controlled amount fields is the currency exchange (CURRENCY_EXCHNG) record. The two amount fields on this record are from amount (FROM_AMT) and converted amount (CONVERTED_AMT). Each has a currency control field of from currency (FROM_CUR) and to currency (TO_CUR), respectively.

3. Ensure that the edit properties of the currency control field are set up correctly.
4. Set the currency control field property of the currency amount record field.

You must associate the field containing the amount with the currency control field.

Open the Record Field Properties dialog box for the currency amount field by right-clicking the record field in the record definition and selecting Record Field Properties.

In the Currency Control Field drop-down list box, select the appropriate currency control field for the currency amount field.

5. Click OK to accept the property settings.
6. Save the record definition.

Setting Currency Field Display Properties When Developing Applications

When you place a currency amount field on a page, you can choose whether to display the currency symbol in the amount field along with the numerical amount and whether to display a thousands separator. Regardless of whether you display the symbol in the field, it is not stored in the database. Currency field formatting is performed only if the Multi-Currency option is selected on the PeopleTools Option page.

To set currency field display properties:

1. In PeopleSoft Application Designer, display the field's Page Field Properties dialog box.

To open the Page Field Properties dialog box for the currency field, right-click the page field and select Page Field Properties.

2. Set the currency display options.

Select the Currency Symbol check box to display the currency symbol (as defined for the currency in the Currency Code table). Clear the check box if you don't want to display the symbol.

Select the 1000 Separator check box if you want the currency amount to appear with a thousands separator character. The specific character that is used as the thousands separator is determined by your international preference.

3. Click OK to accept the dialog box settings.
4. Save the page definition.

See Also

[Chapter 2, “Controlling International Preferences,” Setting Up PeopleSoft Pure Internet Architecture Locale-Based Formatting, page 18](#)

[Chapter 5, “Controlling Currency Display Format,” Using System-Wide Multicurrency Settings, page 63](#)

Using System-Wide Multicurrency Settings

This section provides an overview of multicurrency settings and discusses how to:

- Connect a currency control field to the Multi-Currency check box.
- Enable or disable system-wide currency settings.

Understanding Multicurrency Settings

The Multi-Currency check box on the PeopleTools Options page is a system-wide switch that enables:

- Automatic formatting of those currency amount fields that have associated currency control fields.
- Validation of user entries against the currency’s defined decimal precision.

This causes the system to issue an error if the user attempts to enter a decimal precision that is greater than the precision that is allowed by the currency code definition.

- The display of global currency control fields.

You can design an application so that fields that are specifically related to multiple currencies can appear globally or be hidden, depending on the implementation. Although multicurrency fields are hidden from pages when the multicurrency option is disabled, the fields still exist in the page buffer, so if they are required fields, they must have default values.

- The hiding of all fields.

If you implement the application in an enterprise that uses only a single currency, you can hide all of the fields by clearing the Multi-Currency check box. If the enterprise later begins to use multiple currencies, you can redisplay the multicurrency fields by selecting the check box. Apply this technique only to fields that can be globally hidden without affecting the functionality of the application when working in a single currency. Use this technique for user-operated currency control fields, but not for fields that actually display currency. You also need to set the defaults of the currency display fields to the single currency that is used by the enterprise.

Unless you specifically want to maintain your entire system in a single currency, the Multi-Currency check box remains selected.

Connecting a Currency Control Field to the Multi-Currency Check Box

To connect a currency control field to the system-wide multicurrency check box when developing applications:

1. In PeopleSoft Application Designer, display the field's Page Field Properties dialog box.
To open the Page Field Properties dialog box for the currency field, right-click the page field, and select Page Field Properties. In the Field Properties dialog box, select the Use tab.
2. Select the Multi-Currency Field check box on the Use tab to specify that the field can appear or hide from the PeopleTools Options page.
3. Click OK.
4. Save the page definition.

Enabling or Disabling System-Wide Currency Settings

To enable or disable system-wide currency settings at installation:

1. Select PeopleTools, Utilities, Administration, PeopleTools Options.
The PeopleTools Options page appears.
2. Select or clear the Multi-Currency check box.
Selecting the check box activates automatic currency formatting; clearing the check box disables automatic currency formatting.
If any page fields have the Multi-Currency Field option selected in the Page Field Properties dialog box, selecting the Multi-Currency check box displays those page fields; clearing the check box hides the page fields.
3. Click Save.

Examples

The following examples demonstrate the different outcomes of setting the Multi-Currency Field option in a field's properties and on the PeopleTools Options page to control a field's visibility.

- In the field properties, the Multi-Currency Field check box is *not* selected. On the PeopleTools Options page, the Multi-Currency check box is *not* selected.

In this case, the field's visibility is not controlled by the PeopleTools Options page, so it appears (or doesn't appear) based on the developer coding.

- In the field properties, the Multi-Currency Field check box is *not* selected. On the PeopleTools Options page, the Multi-Currency check box *is* selected.

In this case, the field's visibility is not controlled by the PeopleTools Options page, so it appears (or doesn't appear) based on the developer coding.

- In the field properties, the Multi-Currency Field check box *is* selected. On the PeopleTools Options page, the Multi-Currency check box *is* selected.

In this case, the field's visibility is controlled by the PeopleTools Options page, so it appears.

- In the field properties, the Multi-Currency Field check box *is* selected. On the PeopleTools Options page, the Multi-Currency check box is *not* selected.

In this case, the field's visibility is controlled by the PeopleTools Options page, so it does *not* appear.

Resizing Currency Fields by Using the International Field Size Utility

PeopleSoft applications are typically shipped with amount fields that are sized at 15 integer positions and 3 decimal positions. However, some older applications may be shipped with smaller field sizes, and to use a greater precision, you may need to run the International Field Size utility when you install your PeopleSoft system. If the PeopleSoft installation guide for your product requires it, use the Set International Field Sizes Structured Query Report (SQR) utility (TLSINST1) to increase the length and number of decimal position settings of field definitions. For example, you may need to do this if you process low-value currencies that require numeric fields that are longer than those that are provided in the standard application.

PeopleTools is delivered with the International Field Sizes table (PS_INTL_FLDSIZ_TBL) populated with suggested lengths for numeric fields that may require a large number of digits for certain currencies. You can edit the contents of this table to suit your own requirements, adding and deleting fields and adjusting the new field sizes and decimal positions as necessary.

Keep in mind the limitations on numeric field size and precision imposed by your database platform, and the fact that some numeric fields are used in COBOL and may require COBOL working storage changes in addition to the changes performed by this utility.

To resize international currency fields:

1. Start the PeopleTools International Field Size utility.

Select PeopleTools, Utilities, International, Process Field Size to display the International Field Size page. The International Field Size page shows the name of each field that will be adjusted, the current size of the field, and the proposed new size of the field (stored in the International Field Sizes table).

2. Edit the data in the International Field Sizes table.

Adjust the new field sizes to meet your own requirements. You can insert or delete rows from the page.

3. Save your changes.

4. Run the Process Request Dialog.

To do this: click Run to open the Process Scheduler Request page. Set the report options, and then click OK to run the Set International Field Sizes process.

The utility updates the field definitions and creates a report showing all page fields that have been affected by the changes in field size.

5. Check all affected pages, and rearrange page fields as necessary to rectify overlaps.

The process does not update page field information such as overlapping fields that may be caused by an increase in field length; however, the system automatically adjusts the sizes of any page fields with a size property of *Average* or *Maximum*. If the page field size is *Custom*, it is not adjusted. If the page field is too small, the larger amount can still be entered; however, users may have to scroll to the right to see the full amount, and the truncated amount may be misleading.

6. Rebuild (SQL ALTER) any tables that have been affected by the changes in field sizes.

Use the Application Designer Build feature, with the Alter Tables option, to build any tables (using SQL ALTER) that have been affected by the changes in field size. Use the Find Object References features to determine which tables have been affected, or create a query against the PSRECDEFN, PSRECFIELD, and International Field Sizes tables. For example:

```
SELECT DISTINCT A.RECNAME
FROM PSRECDEFN A, PSRECFIELD B, PS_INTL_FLDSIZ_TBL C
```

```
WHERE A.RECNAME = B.RECNAME  
      AND B.FIELDNAME = C.FIELDNAME  
      AND A.RECTYPE = 0
```

CHAPTER 6

Running COBOL in a Unicode Environment

This chapter provides an overview of COBOL in a Unicode environment and discusses how to:

- Run the COBOL conversion utility.
- Fine-tune COBOL programs.

See Also

[Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” Understanding Character Sets, page 35](#)

Understanding COBOL in a Unicode Environment

This section discusses:

- Unicode encodings in PeopleSoft COBOL.
- Expanded storage space requirements.
- Special logic for parsing Unicode strings.
- COBOL sorting.
- Unicode-specific error messages.

Unicode Encodings in PeopleSoft COBOL

The character set that is used for PeopleSoft COBOL processing must match the character set for your database. If you created a Unicode database for the PeopleSoft system, you must also run COBOL in Unicode.

The Unicode standard provides several methods of encoding Unicode characters into a byte stream. Each encoding has specific properties that make it suitable for use in different environments. The two main encodings that are important to understanding how PeopleSoft COBOL operates when running in Unicode are:

- UCS-2, which is the Unicode encoding that PeopleTools uses internally for data that is held in memory on the application server.

UCS-2 encodes all characters into a fixed storage space of 2 bytes.

- UTF-8, which is the encoding that the PeopleSoft system uses in COBOL.

UTF-8 is a variable format of 1 to 4 bytes per character. Currently, the PeopleSoft system supports only 1- to 3-byte characters. Four-byte UTF-8 characters are required only to represent characters that are outside Unicode’s Basic Multilingual Plane (BMP). In UTF-8, the actual number of bytes to store a character can be determined by the first 3 bits of the first, or only, byte of a character.

The following table shows how the bit pattern of the first byte is related to the number of bytes needed to store the character:

Bit Pattern of First Byte	Character Length
0xxxxxxx	1 byte
110xxxxx	2 bytes
111xxxxx	3 bytes

For example, the following graphic shows the different UCS-2 and UTF-8 encodings for two sample characters:

Character	UCS-2	UTF-8
a	= <u>0x00,0x61</u>	= <u>0x61</u>
学	= <u>0x5B,0x66</u>	= <u>0x5B,0x66,0x22</u>

UCS-2 and UTF-8 encodings

The PeopleSoft system transparently handles the conversion between UCS-2 and UTF-8 when data is passed into the COBOL program from the database. If you are reading or writing files directly from a COBOL program, the input and output files are UTF-8 encoded when running PeopleSoft COBOL programs in Unicode.

Note. In this document, the word *character* refers to a single character in any language, regardless of how many bytes are required to store the character.

Expanded Storage Space Requirements

Moving to a COBOL Unicode environment means that character data can potentially require three times the storage space that is required in a single-character environment, given the variable length of a character that is encoded in UTF-8 from 1 to 3 bytes. To allow for this, all internal data definitions for character-type data in COBOL programs must be expanded to allow for three times as many bytes. This expansion is critical because in a Unicode PeopleSoft database, column sizes are calculated based on a character length, not a byte length. So, a CHAR(10) column on a Unicode database allows the storage of 10 characters, regardless of how many bytes each character takes to store. Given the 3-bytes-per-character maximum requirement of UTF-8 (4-byte UTF-8 characters are not yet supported by the PeopleSoft system), the maximum byte size of this CHAR(10) column is 30 bytes. Therefore, a COBOL type of PIC X(30) may be required to store the contents of a CHAR(10) field on a Unicode database.

The PeopleSoft system provides a COBOL conversion utility to automatically expand the character-data fields in the working storage to accommodate the number of bytes in the UTF-8 encoding scheme.

See Also

Chapter 6, “Running COBOL in a Unicode Environment,” Running the COBOL Conversion Utility, page 70

Special Logic for Parsing Unicode Strings

In a non-Unicode COBOL installation, parsing through a string is easy because you can assume that all characters coming in are 1 byte in length. But in UTF-8, a character can vary between 1 byte and 3 bytes in length. Therefore, you must incorporate special logic to handle string parsing when you are dealing with characters in UTF-8 format.

See Also

[Chapter 6, “Running COBOL in a Unicode Environment,” Identifying Unicode and Non-Unicode Data, page 81](#)

[Chapter 6, “Running COBOL in a Unicode Environment,” Defining Single Character Arrays, page 83](#)

COBOL Sorting

Any in-memory sorting that is performed by using COBOL functions is performed as a binary sort in the current character encoding that is used for COBOL processing and may not necessarily match the sort order that is returned by the database in response to an ORDER BY clause. If you require the database to return data that is sorted by using a binary sort of its encoding rather than the default linguistically correct sort, you must use the %BINARYSORT meta-Structured Query Language (meta-SQL) function around each column in the WHERE or ORDER BY clause where binary ordering is important.

However, for DB2 UDB for OS/390 and z/OS implementations, this binary sorting is equivalent only when the COBOL program is run on a DB2 UDB for OS/390 and z/OS server. For example, the binary sort that is produced in COBOL differs from the binary sort that is produced by the database, because the database is encoded in EBCDIC and the client is in an ASCII-based encoding. Therefore, use the %BINARYSORT meta-SQL function only in COBOL programs that are not run through RemoteCall (the DB2 UDB for OS/390 and z/OS platform is not supported as a RemoteCall server).

When running against non-z/OS systems, the %BINARYSORT function can be used in both RemoteCall and non-RemoteCall programs.

For example:

```
SELECT RECNAME FROM PSRECDEFN WHERE %BINARYSORT(RECNAME) < %BINARYSORT('xxx')
SELECT RECNAME FROM PSRECDEFN ORDER BY %BINARYSORT(RECNAME)
```

Note. Using the %BINARYSORT token in WHERE and ORDER BY clauses often negates the use of any indexes, because most databases can't use indexes for functional comparisons (for example, WHERE %BINARYSORT(column) > 'X'). Use this syntax only when sorting equivalence of SQL statement results and when COBOL memory order is absolutely required.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “Meta-SQL”

Unicode-Specific Error Messages

The following error messages can occur when you are running a COBOL program against a Unicode database:

- Fetch failed: unsuccessful UCS-2 to UTF-8 conversion on column <column number>.
- Bind of parameter failed: unsuccessful UTF-8 to UCS-2 conversion on column <column number>.
- Attempting to use a non-Unicode API to access a Unicode database.
- Attempting to use a non-Unicode COBOL with a Unicode database.

- Attempting to use a Unicode API to access a non-Unicode database.
- Fetch failed: the converted Unicode length of <length> exceeds the allocated buffer length <length> for column <column number>.

These messages appear in the COBOL output log file.

Running the COBOL Conversion Utility

This section provides an overview of the COBOL conversion utility and discusses how to:

- Run the conversion utility.
- Identify converted COBOL programs.
- Understand what is expanded.
- Use utility directives.
- View error logs.

Understanding the COBOL Conversion Utility

As delivered by PeopleSoft, all COBOL programs are configured to run only on non-Unicode databases. To run the PeopleSoft-delivered COBOL on a Unicode database, it first must be converted by using the PeopleTools COBOL conversion utility. This utility is typically called automatically by the PeopleSoft installation process; however in certain circumstances, such as when you adapt COBOL code or apply a PeopleSoft-provided patch to a COBOL program, you may need to run the COBOL conversion utility manually.

Moving to a COBOL Unicode environment means that character data can potentially require three times the storage space that is required in a single-character environment. To allow for this, all internal data definitions for character-type data in COBOL programs must be expanded to allow for three times as many bytes.

Adapt and apply patches to only one set of COBOL source code—non-Unicode source. It is much easier to write COBOL programs without having to remember to triple the size of your working storage as you go. Once your adaptation or patch is complete and you are ready to compile the program, first run it through the COBOL conversion utility, then compile it. This approach has several benefits over customizing the converted code:

- You maintain a single source tree for all of your COBOL—the non-Unicode source.

This way you don't run the risk of accidentally adapting both the non-Unicode COBOL programs *and* the Unicode-converted COBOL programs and potentially losing the modifications to the converted programs the next time you run the converter.

- Although PeopleSoft tests all delivered COBOL programs and patches in both Unicode and non-Unicode environments, only non-Unicode versions of the source are delivered.

Therefore, any time you apply a PeopleSoft COBOL patch to a Unicode system, the patched source code must be run through the COBOL converter. If you had already modified the post-converted source, the reconversion would obliterate your modifications.

If the COBOL conversion utility makes modifications to your code that are undesirable, instead of modifying the postconverted code, the PeopleSoft system provides a series of directives to the utility that can tell it how specific lines of code should (or should not be) converted. This enables you to limit your changes only to the nonconverted code and to make the conversion completely automated.

In a non-Unicode (also known as ANSI) implementation, 1 character typically occupies 1 byte of storage. So for a 10-character field, you can define a PICTURE clause of PIC X(10). In a Unicode implementation, however, you must allow for the maximum number of storage bytes that are required for any character field. Therefore, in the Unicode environment, you must define this same 10-character field with a PICTURE clause of PIC X(30).

To accommodate the number of bytes in the UTF-8 encoding scheme, the PeopleSoft system provides a COBOL conversion utility to expand the character fields in the working storage.

Running the Conversion Utility

Use the following command syntax to run the COBOL conversion utility:

```
<PS_HOME>\bin\client\winx86\pscblucvrt.exe -s:<Source Directory> -t:<Destination=>
Directory> [-r:<TEMP Directory>]
```

Command	Description
-s:<Source Directory>	Specify the source directory where the non-Unicode version of COBOL resides. For the directory, you must specify where the COBOL subdirectories reside (\BASE, \WIN32, \UNIX, and so on). Example: -s:d:\PT8\SRC\CBL
-t:<Destination Directory>	Specify where you want to place the expanded version of COBOL. The utility puts the modified source file in the same COBOL subdirectory in which it was found. Example: -t:d:\PT8\SRC\CBLUNICODE
-r or -rd:<Temp directory>	See Chapter 6, “Running COBOL in a Unicode Environment,” Viewing Error Logs, page 79. -r generates only the summary log file; -rd generates all of the log files.

The utility produces a new source file for each .CBL file that is found. These new files are placed in the PS_HOME\src\ directory.

As delivered by PeopleSoft, the batch utilities that compile your COBOL programs include logic to convert all programs and copybooks before compiling. This logic is triggered only when the Unicode version of PeopleTools is installed.

Compiling COBOL in Microsoft Windows NT or Windows 2000

Use the <PS_HOME>\setup\cbl2uni.bat command to convert all of the COBOL programs and copybooks that are found in the <PS_HOME>\src\cbl directory. After the conversion, <PS_HOME>\src\cbl Unicode contains the expanded COBOL source codes.

Compiling COBOL in UNIX/Linux

Use the <PS_HOME>/install/pscbl.mak command to trigger the conversion before any COBOL programs are compiled. This utility stores all converted programs in the <PS_HOME>/src/cblunicode directory.

Identifying Converted COBOL Programs

When the COBOL conversion utility runs, it places a comment at the beginning of each COBOL program that it converts:

```
***** Converted for Unicode: <date converted>␣
***** DO NOT MODIFY THIS FILE *****␣
**** ALL SOURCE CODE CHANGES MUST BE DONE TO THE ANSI VERSION ****␣
***** FOUND IN %PS_HOME%/SRC/CBL *****
```

COBOL conversion utility comment

This comment line identifies converted programs in two ways:

- A person looking at the program can tell whether it has been converted.
- If you attempt to convert the COBOL source file again, this comment line prevents the conversion utility program from expanding the working storage of this COBOL source file again.

Understanding What Is Expanded

For the utility to recognize when it is appropriate to expand data, strict adherence to the PeopleSoft COBOL coding standards is required. The utility looks for certain code-style patterns to make these decisions.

The conversion utility expands all PIC X[(N)] data fields to triple their original size, with the following exceptions:

- Exception 1: SQL buffer setup data.
- Exception 2: Redefined character fields.
- Exception 3: Fields that appear to be dates.
- Exception 4: Arrays comprising a single character.

The utility also converts copybooks on the fly: the first time that a copybook is referenced inside a code module, it is expanded immediately.

The utility processes an entire set of COBOL modules in a single run. It maintains a record of what it has converted to avoid converting copybooks twice.

Note. The COBOL conversion utility ensures that edited lines do not go past the 72nd column. If the conversion would normally cause a line to exceed that limitation, the utility removes some of the blank spaces between the field name and the PIC X string so that the line fits in the allowed area.

Exception 1: SQL Buffer Setup Data

SQL buffer setup data that refers to the numeric or date data types of SELECT-SETUP or BIND-SETUP is not expanded.

For the interface to PTPSQLRT, a COBOL program passes a SELECT list (SELECT-DATA) and a descriptor area (SELECT-SETUP). The program also passes similar data and setup areas for bind variables. The descriptors that are passed are always character-type data with embedded values that signal the actual data type and length of the data fields. Because these descriptors represent the lengths of the associated data fields in the corresponding SELECT-DATA and BIND-DATA structures, the utility adjusts only the length of the descriptors that are representing character-type data.

Example 1: In the following code, the select list contains two character fields (EMPLID and NAME), a small integer (EMPL_RCD), and a date (EFFDT):

```

SELECT-SETUP.
02 FILLER PIC X(60) VALUE ALL 'C'.
02 FILLER PIC X(2) VALUE ALL 'S'.
02 FILLER PIC X(10) VALUE ALL 'D'.
02 FILLER PIC X(90) VALUE ALL 'C'.
SELECT-DATA.
02 EMPLID PIC X(60).
02 EMPL_RCD PIC 99 COMP.
02 EFFDT PIC X(10).
02 NAME PIC X(90).

```

In Unicode, the only fields that should be expanded are the two character fields (EMPLID and NAME). Numeric data is never affected by Unicode, and (according to the PeopleTools definition), dates are not affected either: they are treated as numeric strings and cannot have special characters.

Thus, the utility converts this code as follows:

```

SELECT-SETUP.
02 FILLER PIC X(60) VALUE ALL 'C'.
02 FILLER PIC X(2) VALUE ALL 'S'.
02 FILLER PIC X(10) VALUE ALL 'D'.
02 FILLER PIC X(90) VALUE ALL 'C'.

SELECT-DATA.
02 EMPLID PIC X(60).
02 EMPL_RCD PIC 99 COMP.
02 EFFDT PIC X(10).
02 NAME PIC X(90).

```

Example 2: The following code represents non-Unicode COBOL (COBOL that has not yet been expanded):

```

01 I-ERRL.
    05 SQL-STMT PIC X(18) VALUE
        'INPXPROC_I_ERRL'.
    05 BIND-SETUP.
        10 FILLER PIC X(10) VALUE ALL 'C'.
        10 FILLER PIC X(6) VALUE 'OPPPPP'.
        10 FILLER PIC X(4) VALUE ALL 'I'.
        10 FILLER PIC X VALUE 'H'.
        10 FILLER PIC X(18) VALUE ALL 'C'.
        10 FILLER PIC X(4) VALUE ALL 'I'.
        10 FILLER PIC X(4) VALUE ALL 'N'.
        10 FILLER PIC X(30) VALUE ALL 'H'.
        10 FILLER PIC X(30) VALUE ALL 'C'.
        10 FILLER PIC X(30) VALUE ALL 'H'.
        10 FILLER PIC X(10) VALUE ALL 'C'.
        10 FILLER PIC X(6) VALUE 'OPPPPP'.
        10 FILLER PIC X(8) VALUE 'OPPPPPPP'.
        10 FILLER PIC X VALUE 'Z'.
    05 BIND-DATA.
        10 TSE-JOBID PIC X(10) VALUE SPACES.
        10 TSE-PROC-INSTANCE PIC 9(10) VALUE ZERO COMP-3.
        10 TSE-SET-NBR PIC 9(6) VALUE ZERO COMP.

```

```

10 TSE-EDIT-TYPE PIC X VALUE SPACE.
10 TSE-FIELDNAME PIC X(18) VALUE SPACES.
10 MESSAGE-SET-NBR PIC 9(5) VALUE ZERO COMP.
10 MESSAGE-NBR PIC 9(5) VALUE ZERO COMP.
10 MESSAGE-PARM PIC X(30) VALUE SPACES.
10 MESSAGE-PARM2 PIC X(30) VALUE SPACES.
10 MESSAGE-PARM3 PIC X(30) VALUE SPACES.
10 BUSINESS-UNIT PIC X(10) VALUE SPACES.
10 TRANSACTION-NBR PIC 9(10) VALUE ZERO COMP-3.
10 SEQ-NBR PIC 9(15) VALUE ZERO COMP-3.
10 FILLER PIC X VALUE 'Z' .

```

The utility converts this code as follows:

```

01 I-ERRL.
    05 SQL-STMT PIC X(54) VALUE
        'INPXPROC_I_ERRL' .
    05 BIND-SETUP.
        10 FILLER PIC X(30) VALUE ALL 'C' .
        10 FILLER PIC X(6) VALUE '0PPPPP' .
        10 FILLER PIC X(4) VALUE ALL 'I' .
        10 FILLER PIC X(3) VALUE ALL 'H' .
        10 FILLER PIC X(54) VALUE ALL 'C' .
        10 FILLER PIC X(4) VALUE ALL 'I' .
        10 FILLER PIC X(4) VALUE ALL 'N' .
        10 FILLER PIC X(90) VALUE ALL 'H' .
        10 FILLER PIC X(90) VALUE ALL 'C' .
        10 FILLER PIC X(90) VALUE ALL 'H' .
        10 FILLER PIC X(30) VALUE ALL 'C' .
        10 FILLER PIC X(6) VALUE '0PPPPP' .
        10 FILLER PIC X(8) VALUE '0PPPPPPP' .
        10 FILLER PIC X VALUE 'Z' .
    05 BIND-DATA.
        10 TSE-JOBID PIC X(30) VALUE SPACES.
        10 TSE-PROC-INSTANCE PIC 9(10) VALUE ZERO COMP-3.
        10 TSE-SET-NBR PIC 9(6) VALUE ZERO COMP.
        10 TSE-EDIT-TYPE PIC X(3) VALUE SPACES.
        10 TSE-FIELDNAME PIC X(54) VALUE SPACES.
        10 MESSAGE-SET-NBR PIC 9(5) VALUE ZERO COMP.
        10 MESSAGE-NBR PIC 9(5) VALUE ZERO COMP.
        10 MESSAGE-PARM PIC X(90) VALUE SPACES.
        10 MESSAGE-PARM2 PIC X(90) VALUE SPACES.
        10 MESSAGE-PARM3 PIC X(90) VALUE SPACES.
        10 BUSINESS-UNIT PIC X(30) VALUE SPACES.
        10 TRANSACTION-NBR PIC 9(10) VALUE ZERO COMP-3.
        10 SEQ-NBR PIC 9(15) VALUE ZERO COMP-3.
        10 FILLER PIC X VALUE 'Z' .

```

Exception 2: Redefined Character Fields

Character fields that are redefined to a numeric field (and group-level fields that contain such character fields) are not expanded. In instances where the redefined field is also redefined as a character field, the original character field and the redefinition that is a character field *are* expanded.

Example 1: In this example, the DB-PIC-PRECIS-CHAR is not expanded:

```
07 DB-PIC-PRECIS-CHAR PIC X(2).
07 DB-PIC-PRECIS-NUM REDEFINES
    DB-PIC-PRECIS-CHAR PIC 9(2).
```

Example 2: In this example, the I-REMIT-ADDR-SEQ is not expanded:

```
02 I-REMIT-ADDR-SEQ PIC 9(04).
02 I-REMIT-ADDR-SEQ-C REDEFINES
    I-REMIT-ADDR-SEQ PIC X(04).
```

Example 3: In this example, the original definition is a character-type field. Although some of the redefined fields are numeric fields, all of the character fields, including the original definition, are expanded.

```
02 MSGDATA1 PIC X(30) VALUE SPACES.
02 FILLER REDEFINES MSGDATA1.
    03 MSGDATA1-INT PIC Z(9)9-.
    03 INT-FILL1 PIC X(19).
02 FILLER REDEFINES MSGDATA1.
    03 MSGDATA1-DOL PIC Z(9)9.99-.
    03 DOL-FILL1 PIC X(16).
02 FILLER REDEFINES MSGDATA1.
    03 MSGDATA1-DEC PIC Z(9)9.9(5)-.
    03 DEC-FILL1 PIC X(13).
```

Exception 3: Fields That Appear to Be Dates

Fields and group-level fields that appear to be dates are not expanded, unless the EXPAND directive is specified for this field or group-level field.

The following table describes the criteria that are used to determine fields or group-level fields as dates:

DATE Data Type	Field or Group-Level Field Name	Field Length or Total Length of a Group-Level Field*
Date	Contains -DT or DATE	10
Time	Contains -TM or TIME	15
Date-Time	Contains -DTTM, DATE, or TIME	26

* When calculating the total length, the utility considers that a group-level field may contain REDEFINE fields. The length of the REDEFINE field is not included when determining the total length of the group field.

Example 1: The field in this example is not expanded:

```
10 START-DATE PIC X(10) VALUE SPACES.
```

Example 2: The fields in this example are not expanded:

```
02 W-EMP-BIRTHDATE.
03 YEAR           PIC 9(4).
03 FILLER         PIC X(1).
03 MONTH         PIC 99.
03 FILLER         PIC X(1).
03 DAYS          PIC 99.
```

Example 3: The fields in this example are not expanded:

```
03 PAY-DATE-TIME.
04 PAY-DTTM-DATE  PIC X(10).
04 PAY-DTTM-DELIM1 PIC X      VALUE '-'.
04 PAY-DTTM-TIME  PIC X(15).
```

Example 4: The fields in this example are not expanded:

```
05 BEGIN-DTTM-TIME.
07 SYS-HOUR       PIC 99      VALUE ZERO.
07 FILLER         PIC X      VALUE SPACE.
07 SYS-MINUTE     PIC 99      VALUE ZERO.
07 FILLER         PIC X      VALUE SPACE.
07 SYS-SECOND     PIC 99      VALUE ZERO.
07 FILLER         PIC X      VALUE SPACE.
07 SYS-MICRO-SECOND PIC 9(6) VALUE ZERO.
```

Example 5: In this example, the group field contains REDEFINE fields. The conversion utility expands the fields because the group field meets the criteria for expansion: it has a total length of 10 and the field name includes the *-DT* string.

```
02 END-DT.
03 END-DT-YY      PIC X(4).
03 END-DT-YY-NUM  REDEFINES END-DT-YY
PIC 9999.
03 FILLER         PIC X.
03 END-DT-MM      PIC XX.
03 END-DT-MM-NUM  REDEFINES END-DT-MM
PIC 99.
03 FILLER         PIC X.
03 END-DT-DD      PIC XX.
03 END-DT-DD-NUM  REDEFINES END-DT-DD
PIC 99.
```

Exception 4: Arrays Comprising a Single Character

For arrays that comprise a single character, the PIC clause is expanded for character data, but the OCCURS clause is not expanded. However, if the data name ends with *-POS*, *-CHAR*, or *-BYTE*, the OCCURS clause is expanded, instead of the element size.

Example 1: In this example, the field is expanded:

```
01 CHAR-ARRAY PIC X      OCCURS 80 TIMES.
   Is expanded to:
01 CHAR-ARRAY PIC X(3)   OCCURS 80 TIMES.
```

Example 2: In this example, the data name ends with *-POS*; therefore, the OCCURS clause is expanded, instead of the element size:

```
01 CHAR-POS PIC X OCCURS 80 TIMES.
    Is expanded to:
01 CHAR-POS PIC X OCCURS 240 TIMES.
```

Using Utility Directives

The COBOL conversion utility accepts various directives in the first six columns of COBOL code. Use these directives to override the utility's normal mode of processing for a single source code line or for a block of lines.

Directive	Description	Purpose
NOCBGN	No conversion: begin	Starting with this line, do not perform expansions.
NOCEND	No conversion: end	End the NOCBGN directive following this line (the directive line is not expanded).
NOCLN	No conversion: line	Do not perform expansions in this single line.
COCCUR	Convert occurrence	Expand the OCCURS clause instead of the PIC clause in this line.
EXPEOF	Expand end of field	Expand a group item by increasing the length of the last field in the group.
EXPAND	Instruct utility to expand field	Force expansion of fields that would normally not be expanded because they appear to be date, time, or datetime fields.

The following examples use existing PeopleSoft COBOL programs to illustrate possible uses for the utility directives.

NOCBGN, NOCEND, and NOCLN Directives

One of the COBOL programs for PeopleSoft Human Resources has a unique way of setting the PAY-PERIODS group field. The program defines an 88-level definition based on the concatenated value of the five, one-column, character-type fields. If the conversion utility were to convert the program without the special directives, none of the cases that are defined in the 88-level field would ever be true.

```
NOCBGN      03  PAY-PERIODS.
              88  PAY-PERIODS-ALL          VALUE 'YYYYY' .
              88  PAY-PERIODS-ALL-BIWEEKLY VALUE 'YYNNN' .
              88  PAY-PERIODS-ALL-SEMIMONTHLY VALUE 'YNNNN' .
```

```

      88 PAY-PERIODS-NONE          VALUE 'NNNNN'.
      04 PAY-PERIOD1             PIC X.
      04 PAY-PERIOD2             PIC X.
      04 PAY-PERIOD3             PIC X.
      04 PAY-PERIOD4             PIC X.
      04 PAY-PERIOD5             PIC X.
      03 FILLER REDEFINES PAY-PERIODS.
      04 PAY-PERIOD              PIC X OCCURS 5.
      88 PAY-PERIOD-YES          VALUE 'Y'.
NOCEBD      88 PAY-PERIOD-NO          VALUE 'N'.
01 S-DEDPDS.
      02 SQL-STMT                 PIC X(18) VALUE
                                'PSPDCFSA_S_DEDPDS'.
      02 BIND-SETUP.
      03 FILLER                   PIC X(10) VALUE ALL 'C'.
      03 FILLER                   PIC X(10) VALUE ALL 'H'.
      03 FILLER                   PIC X(10) VALUE ALL 'D'.
      03 FILLER                   PIC X(10) VALUE ALL 'A'.
NOCEBGN      03 FILLER              PIC X VALUE ALL 'C'.
      03 FILLER                   PIC X VALUE ALL 'H'.
      03 FILLER                   PIC X VALUE ALL 'C'.
      03 FILLER                   PIC X VALUE ALL 'H'.
NOCEBD      03 FILLER              PIC X VALUE ALL 'C'.
      03 FILLER                   PIC X VALUE 'Z'.
      02 BIND-DATA.
      03 COMPANY                  PIC X(10).
      03 PAYGROUP                 PIC X(10).
      03 PAY-END-DT               PIC X(10).
      03 YEAR-END-DT              PIC X(10).
NOCLN      03 PAY-PERIODS          PIC X(5).
      03 FILLER                   PIC X VALUE 'Z'.

```

COCCUR Directive

The conversion utility doesn't normally expand the size of the array in this line from one of the PeopleTools COBOL programs. Using the COCCUR directive ensures that the OCCURS clause is expanded:

```

      02 PARM.
COCCUR      05 PARM-CH              OCCURS 30 TIMES
                                PIC X.

```

EXPEOF Directive

In the following example, the FIELDNAME group-level field is broken down to check the first 4 characters of the string. In this instance, it makes more sense to adjust the length of the FILLER field. By using the EXPEOF directive, you direct the utility to expand the FILLER field to a length of 50:

```

EXPEOF      02 FIELDNAME.
      03 FIELDNAME4               PIC X(4) VALUE SPACE.
      88 FIELDNAME-TSE            VALUE 'TSE_'.
      03 FILLER                   PIC X(14) VALUE SPACE.

```

Viewing Error Logs

The COBOL conversion utility produces a set of error and warning logs with messages that identify nonstandard code styles and inconsistencies. The utility also logs expansion actions that may require manual review.

The utility produces the following logs:

Exception log	This log contains warnings that occurred because of ambiguous working storage definitions. You may need to modify code or add utility directives to resolve the issues logged.
Exception BIND/SELECT log	This log contains warnings and errors that occurred because of ambiguous BIND and SETUP definitions.
Exception date log	This log lists all group-level date fields that are detected by the utility.
Summary log	This log provides general statistics regarding the number of programs that are processed.

When you specify the -4 flag, you see only the summary log. Set the -rd flag on the conversion utility command line if you want the utility to produce all of the detail logs: exception, BIND/SELECT, and exception date.

Messages from the Exception Log

The following tables summarize all of the messages that can appear in the three exception log files. Errors indicate problems that are encountered by the conversion utility.

Message	Type	Note
Non-matching conversion block found in line <line number>.	Error	Detected the NOCBGN directive, but couldn't find the corresponding NOCEND.
Error in determining numeric length in line <line number>.	Error	Couldn't decipher the numeric PICTURE clause.
The size of the one character array will be expanded in line <line number>.	Warning	Detected a 1-character array where the field contains the string -BYTE, -POS, or -CHAR.
A one-character array is found in <line number>. The conversion routine will expand this to PIC X(3).	Warning	None.
Unable to find the copy library <copy library name>.	Error	Couldn't locate the copy library file that the program references.

Messages from the Exception BIND/SELECT Log

The following table lists messages from the BIND/SELECT log:

Message	Type	Note
Didn't find the corresponding DATA section for <DATA field name> in line <line number>.	Error	Detected either a BIND-DATA or a SELECT-DATA, but cannot find the SETUP group field.
No delimiter found on <group field name> section in line <line number>.	Warning	Didn't find a FILLER field with a value Z in a DATA or SETUP group field.
Unable to convert the <group field name> section due to problems reading the Copylib.	Error	Couldn't locate a copy library that DATA or SETUP references.
The <group field name> found in line <line number> has a mismatch count.	Warning and error	The number of columns in DATA doesn't match the count for the corresponding SETUP.
Incompatible date type match for field in line <line number>.	Error	The data type definition in SETUP doesn't correspond to the data type in DATA.

Messages from the Exception Date Log

The following table lists messages from the exception date log:

Message	Type	Note
Date/time/datetime detected and will not be expanded in line.	Warning	None.
Verify if a date/time/datetime field in line number: <line number>.	Warning	Found a character-type field or group field with a total length of 10, 15, or 26 that could be a date, time, or datetime.

Fine-Tuning COBOL Programs

Although the COBOL conversion makes most of the changes that are needed to run COBOL in a Unicode environment, some manual fine-tuning may still be necessary.

This section discusses how to:

- Identify Unicode and non-Unicode data.
- Specify column lengths in dynamic SQL.

- Define single character arrays.

Identifying Unicode and Non-Unicode Data

A COBOL program may need to determine whether it's dealing with non-Unicode or Unicode data. For example, if the program parses a string character, it must apply different logic depending on whether the string is non-Unicode or Unicode. The program can get this information from the ENCODING-MODE-SW in the PTCSQLRT copy library (ANSI-Mode is the same as non-Unicode):

```

03  ENCODING-MODE-SW    PIC X(3)    VALUE SPACE.
                                88  ANSI-MODE                VALUE 'A'.
                                88  UNICODE-MODE             VALUE 'U'.

```

The ENCODING-MODE-SW value is set by the COBOL application programming interface (API), which determines which type of data it's dealing with by checking the value of the UNICODE_ENABLED field in the PSSTATUS table. When the value of the UNICODE_ENABLED flag is set to 1 (TRUE), this signals the COBOL API that it is accessing a Unicode database.

The following graphic shows the PeopleSoft COBOL trace file, which provides information about the encoding scheme that the COBOL API used:

PeopleSoft Batch Timings Report: cobsql_0416165133.trc								
Encoding Scheme Used: Unicode								
PeopleSoft Batch Statistics (All timings in seconds)								
Statement	Retrieve Count	Retrieve Time	Compile Count	Compile Time	Execute Count	Execute Time	Fetch Count	Fetch Time
APIBNN	0	0.00	0	0.00	266043	4.71	0	0.00
APISSB	0	0.00	0	0.00	1315	0.00	0	0.00
COMMIT	0	0.00	0	0.00	10	0.22	0	0.00
CONNECT	0	0.00	0	0.00	75	0.92	0	0.00
DISCONNECT	0	0.00	0	0.00	75	1.91	0	0.00

Sample trace file

The COBOL API also performs the necessary translations between the UTF-8 encodings that are required by COBOL and the UCS-2 encodings that are used elsewhere in the PeopleSoft system.

Specifying Column Lengths in Dynamic SQL

Perhaps the biggest effort in getting COBOL fully functional in a Unicode environment is setting up the bind parameters and select buffers of any dynamic SQL statements.

Programs that use dynamic SQL must specify the column lengths for bind or select fields before calling the PTPDYSQL program. Within a COBOL program, there are two ways that you can assign bind parameters and select buffers of dynamic SQL statements:

- By using a predefined working storage area with the dynamic SQL statement.

This method is similar to the method that is used for stored SQL statements. In this case, PTPDYSQL adjusts the length of character-data fields that are passed to PTPSQLRT. This is necessary because the COBOL Unicode conversion utility expands only the working storage fields; it does not modify the length of fields that are hard-coded in the PROGRAM-DIVISION section of the COBOL programs.

Because PTPDYSQL sends the correct length to PTPSQLRT, no changes to the COBOL program are necessary.

- By using a buffer array.

At runtime, this array is partitioned based on the properties of all of the fields that are referenced by the dynamic SQL statement. The properties of those fields are retrieved from the PSDBFIELD table, and include both the field's data type and the field's length.

In this case, you must modify the COBOL program to adjust the length that is specified for a character field. Adjust the length by a factor of three.

To adjust the length of the character field appropriately, the program must recognize the encoding scheme that is used by the COBOL API. The program can take advantage of the ENCODING-MODE-SW field in PTCSQLRT to determine when the length of the field needs to be adjusted.

The following example illustrates the use of a buffer array to calculate the length of a character field in the Unicode environment:

```

02  SELECT-DATA.
      03  FIELDNAME          PIC X(54)  VALUE SPACE.
      03  FIELDNUM          PIC 9(3)    VALUE ZERO  COMP.
      03  DEFFIELDNAME     PIC X(90)   VALUE SPACES.
      03  FIELDLEN         PIC 9(3)    VALUE ZERO  COMP.
      03  FIELDTYPE        PIC 9(2)    VALUE ZERO  COMP.
          88  RDM-CHAR          VALUE ZERO.
          88  RDM-LONG-CHAR     VALUE 1.
          88  RDM-NUMBER        VALUE 2.
          88  RDM-SIGNED-NUMBER VALUE 3.
          88  RDM-DATE          VALUE 4.
          88  RDM-TIME          VALUE 5.
          88  RDM-DATETIME      VALUE 6.
      03  DECIMALPOS        PIC 9(2)    VALUE ZERO  COMP.
      03  FILLER            PIC X      VALUE 'Z'.
          88  RDM-NUMBER        VALUE 2.
          88  RDM-SIGNED-NUMBER VALUE 3.
          88  RDM-DATE          VALUE 4.
          88  RDM-TIME          VALUE 5.
          88  RDM-DATETIME      VALUE 6.
      03  DECIMALPOS        PIC 9(2)    VALUE ZERO  COMP.
      03  FILLER            PIC X      VALUE 'Z'.
      . . .
MOVE CORR SELECT-DATA OF S-RECFLD
      TO FLD-ARRAY OF RECFLD (FLD-IDX)
      *  CONVERT FIELD TYPE FROM PSDBFIELD TYPE TO SQLRT CODE.
MOVE FIELDLEN OF S-RECFLD TO SETUP-LENGTH OF RECFLD (FLD-IDX)
IF RDM-CHAR OF S-RECFLD
      SET SETUP-TYPE-CHAR OF RECFLD (FLD-IDX) TO TRUE
      IF UNICODE-MODE OF SQLRT
          COMPUTE SETUP-LENGTH OF RECFLD (FLD-IDX) =
              SETUP-LENGTH OF RECFLD (FLD-IDX) * 3
      END-IF
END-IF

```

See Also

[Chapter 6, “Running COBOL in a Unicode Environment,” Identifying Unicode and Non-Unicode Data, page 81](#)

Defining Single Character Arrays

Some COBOL programs define single-character arrays to parse or examine a string of characters, one character at a time. In a Unicode environment, be sure that you’re examining the string one character at a time, not one byte at a time.

The following is an example of a code fragment in which the program is examining a string one character at a time:

```

01 CHAR-ARRAY.
   02 CHAR-POS    PIC X    OCCURS 256 TIMES
      INDEXED BY CHAR-IDX.
   88 FIELD-DELIM      VALUE '*' .
. . .
. . .
SET CHAR-IDX TO 1
SEARCH CHAR-ARRAY
  WHEN FIELD-DELIM(CHAR-IDX)
    SET W-OFFSET TO CHAR-IDX
    DISPLAY 'FIELD DELIMITER FOUND AT POSITION ' W-OFFSET
END-SEARCH

```

The intent of the code in the previous example is to examine each character of the array, looking for the first delimiter character. When that character is found, the code displays the position of the delimiter.

In a non-Unicode environment that uses only the Latin1 character set, this works because there is 1 byte (1 array element) per character. In a Unicode environment (or in a non-Unicode environment that allows double-byte character sets), this fails because what could potentially be examined is the second or third byte of a 2- or 3-byte character. It is possible for the second or third byte to match the bit pattern of the delimiter character, thus falsely passing the test and ending the search loop.

To correct this situation, you must know the length (in bytes) of each character that is being processed. A new COBOL function, PTPSTRFN, is available that returns the length of a character so that the code can take this into account when performing a character search. The PTPSTRFN subroutine works for both Unicode character sets and ANSI double-byte characters sets.

The PTPSTRFN subroutine offers two ways of retrieving the character length of a character database type field:

- By requesting the length of a single character.
- By requesting a map of an entire character string.

Choose this option if the application program needs to get the length information of all characters within a string.

Requesting the Length of a Single Character

The input parameters are:

Parameter	Value	Notes
ACTION-TYPE	ACTION-CHARLEN	None.
CHAR-CD	The character whose length you want to verify.	This variable is included in the PTCSTRFN.CBL copy library.

The values that are returned are:

Parameter	Value	Notes
CHAR-LENGTH	The subroutine returns one of the following values, representing the length of the character that is referenced by CHAR-CD: <ul style="list-style-type: none"> • ONE-BYTE • TWO-BYTES • THREE-BYTES 	This variable is included in the PTCSTRFN.CBL copy library.
STRFN-RC	Returns one of the following values: <ul style="list-style-type: none"> • STRFN-RC-OK • STRFN-INVALID-ACTION 	This variable is included in the PTCSTRFN.CBL copy library.

At the beginning of this section, there was a code fragment in which the program was examining a string, one character at a time, looking for the first delimiter character:

```

01 CHAR-ARRAY.
   02 CHAR-POS    PIC X OCCURS 256 TIMES
       INDEXED BY CHAR-IDX.
   88 FIELD-DELIM    VALUE '*'.
. . .
. . .
SET CHAR-IDX TO 1
SEARCH CHAR-ARRAY
  WHEN FIELD-DELIM(CHAR-IDX)
    SET W-OFFSET TO CHAR-IDX
    DISPLAY 'FIELD DELIMITER FOUND AT POSITION ' W-OFFSET
END-SEARCH

```

After the code is modified for Unicode, it looks like this:

```

01 CHAR-ARRAY.
   02 CHAR-POS    PIC X OCCURS 256 TIMES
       INDEXED BY CHAR-IDX.
   88 FIELD-DELIM    VALUE '*'.
01 STR-FUNC      COPY 'PTCSTRFN'.
. . .

```

```

. . .
SET CHAR-IDX TO 1
SEARCH CHAR-ARRAY
    WHEN FIELD-DELIM(CHAR-IDX)
        SET W-OFFSET TO CHAR-IDX
        DISPLAY 'FIELD DELIMITER FOUND AT POSITION ' W-OFFSET
    WHEN OTHER
        MOVE CHAR-POS(CHAR-IDX) TO CHAR-CD OF STR-FUNC
        CALL 'PTPSTRFN' USING ACTION-CHARLEN

STR-FUNC
IF TWO-BYTES OF STR-FUNC
    SET CHAR-IDX UP BY 1
ELSE IF THREE-BYTES OF STR-FUNC
    SET CHAR-IDX UP BY 2
END-IF
END-SEARCH

```

The modification ensures that the code continues to function properly in a Unicode environment. However, we can be sure that modification works only when the character for which the program is searching is 1 byte in length.

Consider the following code fragment, where the character that the program is searching for may be longer than 1 byte:

```

01 W-DELIMITER PIC X(3) VALUE '<some extended character>'.
01 CHAR-ARRAY.
    02 CHAR-POS PIC X OCCURS 256 TIMES
        INDEXED BY CHAR-IDX
            CHAR-IDX2
            CHAR-IDX3.
    88 FIELD-DELIM VALUE '*'.

. . .
. . .
SET CHAR-IDX TO 1
SEARCH CHAR-ARRAY
    WHEN CHAR-POS(CHAR-IDX) = W-DELIMITER
        SET W-OFFSET TO CHAR-IDX
        DISPLAY 'FIELD DELIMITER FOUND AT POSITION ' W-OFFSET
END-SEARCH

```

For this code to work in a Unicode environment, a Unicode-specific search algorithm must be used. Ensure that the program always compares the correct bytes from the array (up to 3 bytes, based on the current character length) to the fixed 3-byte field containing the search value.

The proper search method looks like this:

```

01 CHAR-ARRAY.
    02 CHAR-POS PIC X OCCURS 256 TIMES
        INDEXED BY CHAR-IDX.
    88 FIELD-DELIM VALUE '*'.
01 STR-FUNC COPY 'PTCSTRFN'.

. . .
. . .

```

```

SET CHAR-IDX TO 1
PERFORM UNTIL CHAR-IDX > 256
  MOVE CHAR-POS(CHAR-IDX) TO CHAR-CD OF STR-FUNC
  CALL 'PTPSTRFN' USING ACTION-CHARLEN
                        STR-FUNC

  INITIALIZE W-WORK
  EVALUATE TRUE
    WHEN ONE-BYTE OF STR-FUNC
      MOVE CHAR-POS(CHAR-IDX) TO W-WORK
      SET CHAR-IDX UP BY 1
    WHEN TWO-BYTES OF STR-FUNC
      SET CHAR-IDX2 TO CHAR-IDX
      SET CHAR-IDX2 UP BY 1
      STRING CHAR-POS(CHAR-IDX)
             CHAR-POS(CHAR-IDX2)
             DELIMITED BY SIZE
             INTO W-WORK
      END-STRING
      SET CHAR-IDX UP BY 2
    WHEN THREE-BYTES OF STR-FUNC
      SET CHAR-IDX2 TO CHAR-IDX
      SET CHAR-IDX2 UP BY 1
      SET CHAR-IDX3 TO CHAR-IDX
      SET CHAR-IDX3 UP BY 2
      STRING CHAR-POS(CHAR-IDX)
             CHAR-POS(CHAR-IDX2)
             CHAR-POS(CHAR-IDX3)
             DELIMITED BY SIZE
             INTO W-WORK
      END-STRING
      SET CHAR-IDX UP BY 3
    WHEN OTHER
      DISPLAY '**ERROR** INVALID CHARACTER LENGTH'
      <ABEND>
  END-EVALUATE
  IF W-WORK = W-DELIMITER
    SET W-OFFSET TO CHAR-IDX
    DISPLAY 'FIELD DELIMITER FOUND AT POSITION ' W-OFFSET
  END-IF
END-PERFORM

```

As you can see from the previous example, searching a string array for a particular value that may be an extended character can be difficult; if possible, avoid such a search.

Requesting a Map of an Entire Character String

The input parameters are:

Parameter	Value	Notes
ACTION-TYPE	ACTION-STRMAP.	None.
STRING-LENGTH	Length of the character string <String Parameter 1>.	This variable is included in the PTCSTRFN.CBL copy library.
<String Parameter 1>	The character string.	None.
<String Parameter 2>	The buffer area to be updated by the subroutine.	None.

The values that are returned are:

Parameter	Value	Notes
<String Parameter 2>	This buffer is updated with the appropriate values. This field contains at least one of these values: <ul style="list-style-type: none"> • 1. The next character position is part of a 1-byte character. • 2X. The next 2 character positions are part of a 2-byte character. • 3XX. The next 3 character positions are part of a 3-byte character. 	Refer to the example following this table to see how the function works.
STRFN-RC	Returns one of the following values: <ul style="list-style-type: none"> • STRFN-RC-OK • STRFN-INVALID-ACTION 	This variable is included in the PTCSTRFN.CBL copy library.

The PTPSTRFN COBOL subroutine provides an option to map the byte length of an entire character string. The following sample COBOL code provides an example of how this feature is used. The following graphic shows the Unicode table from which the code reads all fields:

LANGUAGE	UNITEXT
Catalan	Quan el món vol conversar, parla Unicode
Chinese (Simplified)	当世界需要沟通时, 请用Unicode!
Chinese (Traditional)	當世界需要溝通時, 請用統一碼(Unicode)
Danish	Når verden vil tale, taler den Unicode
Dutch	Als de wereld wil praten, spreekt hij Unicode
English	When the world wants to talk, it speaks Unicode
Esperanto	Kiam la mondo volas paroli, ĝi parolas Unicode
Finnish	Kun maailma haluaa puhua, se puhuu Unicodea
French	Quand le monde veut communiquer, il parle en Unicode

Sample Unicode table

For each row in the table, the code performs the following functions:

- Calls PTPSTRFN to get the string mapping of the UTF-8 character string of the text that is retrieved for the UNITEXT field.
- Displays the UTF-8 string equivalent of the text.
- Tallies the number of 1-byte, 2-byte, and 3-byte characters of the text.

The COBOL code analyzes the text as follows:

```

01 W-WORK.
   02 LANGUAGE                PIC X(20).
   02 UNICODE-TEXT            PIC X(300).
   02 UNICODE-TEXT-MAP        PIC X(300).
   02 DATA-LEN                PIC 9(3)    COMP.
   02 BYTE-POS-MAX            PIC 9(4)    COMP.
   02 COUNTERS.
       05 COUNT-1BYTE-CHAR    PIC 9(02)   VALUE ZEROS.
       05 COUNT-2BYTE-CHAR    PIC 9(02)   VALUE ZEROS.
       05 COUNT-3BYTE-CHAR    PIC 9(02)   VALUE ZEROS.
01 BYTE-ARRAY.
   02 BYTE-POS                PIC X    OCCURS 300    TIMES
                               INDEXED BY BYTE-IDX.
       88 ONE-BYTE-CHAR        VALUE '1'.
       88 TWO-BYTES-CHAR       VALUE '2'.
       88 THREE-BYTES-CHAR     VALUE '3'.
       88 BYTE-STRING-END     VALUE SPACE.
01 STR-FUNC                    COPY 'PTPSTRFN'.
. . .
. . .
<Code to retrieve the text from the database and assign to the appropriate=>
fields>
. . .
. . .
* Initialize the string map before calling the function
  MOVE SPACES    TO UNICODE-TEXT-MAP
  CALL 'PTPSTRFN' USING ACTION-STRMAP

```

```

                                STR-FUNC
                                UNICODE-TEXT
                                UNICODE-TEXT-MAP
IF NOT STRFN-RC-OF OF STR-FUNC
    <ABEND PROGRAM>
END-IF
SET BYTE-POS-MAX TO 300
MOVE 300 TO DATA-LEN OF W-WORK
MOVE UNICODE-TEXT-MAP TO BYTE-ARRAY
PERFORM VARYING BYTE-IDX FROM 300 BY -1
    UNTIL BYTE-IDX <= 1
    OR NOT BYTE-STRING-END(BYTE-IDX)
    SUBTRACT 1 FROM DATA-LEN OF W-WORK
END-PERFORM
* Initialize counters
MOVE ZEROS TO COUNT-1BYTE-CHAR
MOVE ZEROS TO COUNT-2BYTE-CHAR
MOVE ZEROS TO COUNT-3BYTE-CHAR
PERFORM UNTIL BYTE-IDX > DATA-LEN OF W-WORK
    EVALUATE TRUE
        WHEN ONE-BYTE-CHAR
            ADD 1 TO COUNT-1BYTE-CHAR
        WHEN TWO-BYTES-CHAR
            ADD 1 TO COUNT-2BYTE-CHAR
        WHEN THREE-BYTES-CHAR
            ADD 1 TO COUNT-3BYTE-CHAR
    END-EVALUATE
    SET BYTE-IDX UP BY 1
END-PERFORM
DISPLAY ' LANGUAGE = ' LANGUAGE
DISPLAY ' UTF8 TEXT: (LENGTH = ' DATA-LEN ') '
DISPLAY ' ' UNICODE-TEXT
DISPLAY ' '
DISPLAY ' UTF8 BYTE MAPPING: '
DISPLAY ' ' UNICODE-TEXT-MAP
DISPLAY ' '
DISPLAY ' TALLY: '
DISPLAY ' NUMBER OF ONE-BYTE CHAR: '
    COUNT-1BYTE-CHAR
DISPLAY ' NUMBER OF TWO-BYTES CHAR: '
    COUNT-2BYTE-CHAR
DISPLAY ' NUMBER OF THREE-BYTES CHAR: '
    COUNT-3BYTE-CHAR
DISPLAY ' '
DISPLAY ' '

```

The output of this program appears as follows. Note that certain strings appear to be garbled. This is because there is no font to display the characters properly.

```

LANGUAGE = Catalan
UTF8 TEXT: (LENGTH = 0041)

```


CHAPTER 7

Sorting in PeopleTools

This chapter provides an overview of sort orders and discusses how to:

- Set the sort order.
- Force a binary sort in Structured Query Language (SQL).
- Sort in COBOL.
- Sort in Structured Query Reports (SQR).

Understanding Sort Orders

This section discusses:

- Sorting overview.
- PeopleTools sorts.
- Database-level SQL ORDER BY sorts.
- PeopleTools in-memory sorts.
- Binary sorts.

Sorting Overview

Sorting data in English is reasonably simple given the well-defined sorting rules of the language. Additionally, most character sets are based on the ASCII standard, which allocates characters to numerical codes in English alphabetical order. Therefore, when sorting ASCII data by its binary representation, you automatically get a sort that makes sense in English; data is sorted from A to Z and numerics are sorted from 0 to 9.

However, sorting non-English languages is significantly more complex; some languages have special rules for sorting characters with diacritic marks; others, such as Japanese and Thai, can be sorted in several different orders depending on the usage or context of the sort.

In linguistic circles, sorting is also known as *collation*. In this book, the two terms are used interchangeably.

A sort order identifies how PeopleTools assembles, compares, and displays data. For example, a sort order specifies whether A is less than, equal to, or greater than Z. The simplest way of sorting data in a computer system is to sort it in the order that the characters appear in the character set. This is known as a *binary sort*, because it sorts the numerical codes of each character as they are stored in memory without any special sorting cases or linguistic considerations. A binary sort works well for sorting English language data; English sorting rules can be implemented as a binary sort as long as the underlying character set is laid out from A to Z. English characters in US-ASCII, EBCDIC, and Unicode are all laid out in this fashion, so a binary sort on data in any of these character sets is sufficient for sorting English data.

However, when sorting data in languages other than English, you must consider how to sort:

- Characters with diacritic marks (such as á, ñ, and ö): after the letter Z, or after the base character without the diacritic.

In most character sets, these characters appear after the letter Z in binary order; however, in most languages they sort after the base form of the character.

- Special characters and ligatures (such as æ and œ) and representative characters (such as ß) in some European languages.

In most cases, these characters must be expanded to their full form before being sorted. For example, æ is often expanded to *ae* when sorted.

- Ideographic languages.

The Chinese, Japanese, and Korean languages use a large repertoire of characters in their written languages, such that it would be impossible for the average person to remember an arbitrary sort order. Instead, several schemes exist for sorting Chinese, Japanese, and Korean characters, including sorting based on core, common parts of each character (radicals), or by counting the number of brush or pen strokes that it takes to write the character (stroke count).

Unfortunately, most of these sorting schemes are language-specific and sometimes even specific to a country in which a language is spoken. For example, the German sharp-S character (ß) is sorted in Germany as if it were written as *SS*, but in Austria it is sorted as if it were written as *SZ*. Other times, multiple sort orders can be in use in a single country. In Spain, it is common to sort the *ch* character sequence after *cz* but before *d*. However, in more recent times many Spanish organizations have reverted to sorting the *ch* sequence as individual characters between *cg* and *ci*. Which is correct depends on personal or organizational preference.

In the majority of cases where data is sorted in PeopleTools, the host database management system performs the sort through an ORDER BY clause in SQL, and the result is displayed directly to the user. It is therefore critical, when you create a database, that you select an appropriate sort order based on the languages that you plan to maintain in the database, the countries in which you plan to operate, and any specific preferences or policies that your organization maintains relating to sorting (such as which Spanish *ch* sort is preferred). Most database systems require you to choose a sort order during database creation, because it affects the way that SQL indexes are stored on disk to optimize sorting performance.

Note. Due to limitations in most database packages and for performance reasons, PeopleTools 8.45 supports only one sort order per database.

In some cases, PeopleTools also sorts data in memory and must use internal tables to determine the appropriate order for character data. This is particularly prevalent when sorting drop-down lists on pages displaying translate values and within PeopleSoft Query. It is therefore important that you not only create the database with the appropriate sort order, but you also tell PeopleTools which sort order you have chosen for the database so it can emulate this sort for lists that it sorts in memory.

Note. Sorting in PeopleTools is case sensitive (for example, the lowercase letter *a* sorts after the uppercase letter *Z*) and accent sensitive (for example, the accented letter *á* is considered distinct from the unaccented letter *a*). Sorting in PeopleTools is also kana sensitive for Japanese data, meaning that certain forms of Japanese characters (Hiragana and Katakana) are considered distinct rather than equivalent. PeopleTools does not support case-insensitive, accent-insensitive, or kana-insensitive sorting.

PeopleTools Sorts

Many different components of PeopleTools sort character data, most of which rely on the database system to perform sorting by using a SQL ORDER BY clause. However, PeopleTools does perform some sorting in memory.

See [Chapter 7, “Sorting in PeopleTools,” PeopleTools In-Memory Sorts, page 96.](#)

Other parts of PeopleTools support only a binary sort for logistical or performance reasons. The following table indicates each of the common areas of PeopleTools that presents sorted lists of data to the user and the mechanism that each uses to perform the sort:

Functional Area	Sorting Engine
Component search dialog boxes.	Database-level SQL ORDER BY sort.
Scroll buffers.	Database-level SQL ORDER BY sort.
SQL operations in PeopleSoft Query, PS/n Vision, Crystal, and SQR.	Database-level SQL ORDER BY sort.
SQL functions that are called from PeopleCode (SQL Objects, ExecSQL) containing BETWEEN, <, >, MIN, MAX, and so on.	Database-level SQL ORDER BY sort.
PeopleCode ScrollSelect() / ScrollSelectNew(), RowScrollSelect(), and RollScrollSelectNew() functions.	Database-level SQL ORDER BY sort.
Column-specific re-sorts in PeopleSoft Pure Internet Architecture (by clicking on a column heading).	PeopleTools in-memory sort.
PeopleCode SortScroll().	PeopleTools in-memory sort.
Drop-down list boxes of translate values.	PeopleTools in-memory sort.
PeopleCode binary comparison functions (<, >, =).	Binary sort.
%BINARYSORT meta-SQL functions.	Binary sort.
Greater than and less than COBOL operations.	Binary sort.
Greater than and less than comparisons in SQR.	Binary sort.

Database-Level SQL ORDER BY Sorts

PeopleTools relies on the sorting capabilities of the host database system for each functional area that is identified in the table in the previous section as using a database-level SQL ORDER BY sort. In this case, the SQL statement that is used to retrieve the data to be displayed to the user is coded to include an ORDER BY statement, and the sort order of the database determines in what order data is retrieved. Obviously, this relies on the database system being correctly configured for the appropriate linguistic sort that you determined is required for your database.

Each database management system has a different mechanism of determining the sort order for SQL ORDER BY statements. For example, Microsoft SQL Server requires the user to select a sort order when the server is installed, or when each database is created. Oracle enables the database administrator to specify the appropriate sort order in the init.ora parameter file.

See *PeopleTools 8.45 Installation Guide for Your Platform*

See Also

Documentation delivered by your database vendor

PeopleTools In-Memory Sorts

To avoid round-trips to the database, PeopleTools performs some sorting in memory, typically of short lists, such as a list of translate values. PeopleTools supports a significant number of linguistic sorts for in-memory sorting. The sort order that is used for PeopleTools in-memory sorting is controlled by the sort order option on the PeopleTools Options page. While PeopleTools cannot emulate exactly each sort order that is offered by your database vendor, it provides an appropriate sort order for most popular business languages. Select the sort order that most closely corresponds to the sort order of your database. Conflicting sort order selection between the database and PeopleTools results in lists of values that are sorted by PeopleTools appearing in a significantly different order than lists of values that are sorted by the database.

See [Chapter 7, “Sorting in PeopleTools,” Setting the Sort Order, page 97](#).

Binary Sorts

When comparing values in PeopleCode, SQR, and COBOL that use less than (<), greater than (>), or other character comparison operators, a binary sort is used. The specific sort that is performed depends on the character set of data in memory at the time. For example, all binary sorts that are performed in PeopleCode, SQR syntax, and Unicode COBOL take place based on the Unicode binary order. Binary sorts that are performed in non-Unicode COBOL take place in the non-Unicode character set of the batch server. Binary sorts that are performed by the %BINARYSORT PeopleCode meta-SQL function take place in the character set of the database engine.

Therefore, you should be careful not to write code that depends on the sorting of data by using binary operators matching the sorting of data by the database as the result of sorting operators in SQL statements.

For example, the following PeopleCode statement is performed as a Unicode binary comparison, and *á* is located in the Unicode tables after the character *z*, so it always returns *True*:

```
if 'z' < 'á' then. . .
```

However, when the same comparison is run in SQL (as in the following example), the database's sort order determines which character is greater. In a database that is configured for binary sorting, it returns *true*; however, in a database that is configured for French sorting (where *á* is sorted after *a* but before *b*), it returns *false*.

```
SELECT . . .
WHERE 'z' < 'á'
```

It is important that these functional areas use binary sorting instead of attempting to perform a linguistically-sensitive sort because:

- Any linguistically-sensitive sort that is performed in memory can be only an approximation of the sort that the database system would perform in the same situation given the large number of sort orders that are provided by database vendors and the significant variations to these orders in minor versions of the database software.

Do not assume that a binary sort that is performed in PeopleTools will match the sort of the same characters in a SQL statement that uses the less than, greater than, or BETWEEN operators.

- Performing a guaranteed database-compatible sort for each comparison in PeopleCode would require a round-trip to the database to perform the sorting and would affect the performance of PeopleCode operations.

Setting the Sort Order

As noted previously in this chapter, some components of PeopleTools cannot rely on the database to sort data and must do so in memory. The sort order option on the PeopleTools Options page enables you to select which sort order should be used by PeopleTools when sorting data in memory.

Generally, you should set this option soon after you have completed the installation of the database; choose the option that most closely approximates the sort order that you selected when creating the database.

To set the sort order:

1. Select PeopleTools, Utilities, Administration, PeopleTools Options.
2. Select an option from the Sort Order Option drop-down list box.
3. Click Save.

Note. Due to a database limitation, only binary sort order is supported for Sybase Unicode databases.

Forcing a Binary Sort in SQL

When writing PeopleCode and other procedural logic, it is sometimes expected that sorting a list of data in memory produces the same results as sorting the same list in the database through a SQL ORDER BY statement. When working in some languages, such as English, whose sorting logic is relatively simple, this equivalence may be taken for granted.

However, when running PeopleTools against a database that is using a linguistic sort, it is likely that a greater than, less than, or between comparison of values in memory may produce different results than if the same comparison were performed by the database.

Take, for instance, the following PeopleCode syntax:

```

If START_NAME < END_NAME then
    Error("Start name must be less than end name");
End-If

```

Because all character comparisons in PeopleCode are performed based on the binary value of the character in Unicode (and not on the linguistic weight of the character), this code may produce unexpected results in languages where binary sorting is not sufficient. For example, if START_NAME had a value of *Über* and END_NAME had a value of *Zeifang*, this code produces the error as in the Unicode character set, the uppercase *U* with umlaut (*Ü*) appears after the uppercase *Z*. However, in a German sort, *Ü* should precede *Z*. If the database were created with a German sort order, this would be reflected by the database's sort if the same statement were reflected in SQL that is called from PeopleCode:

```

SQLExec("SELECT 'X' FROM PSLOCK WHERE :1 < :2"
        ,START_NAME, END_NAME, &X);

If &X != 'X' then
    Error("Start name must be less than end name");
End-If

```

The example shows that when you use a linguistically sorted database, these string comparisons may return different results when they are run through the database by using SQL than when they are performed in PeopleCode. Of course numerical values and dates always sort equivalently—this behavior is limited to the sorting of characters and character strings.

In some situations, you may want to disable the linguistic sort that is performed by the database on a statement-by-statement level and have the database perform the comparison at a binary level.

To force a SQL query to return sort-sensitive results based on a binary sort instead of a linguistic sort, a meta-SQL token, %BINARYSORT, is provided. Use this token to wrap each column in an ORDER BY, less than, greater than, or BETWEEN operation where a binary comparison is required. For example, to return all employee names, ordered by last name in binary ordering, use the following SQL:

```
SELECT NAME FROM PS_PERSONAL_DATA
ORDER BY %BINARYSORT(NAME)
```

This may be useful if you are building an ordered array of names in memory that you plan to parse or manipulate with PeopleCode less than or greater than operators.

Similarly, to fetch a list of names from the database where the NAME field is greater than START_NAME by using a binary comparison that is parallel to that used in PeopleTools, use the following syntax:

```
SELECT NAME FROM PS_PERSONAL_DATA
WHERE %BINARYSORT(NAME) < %BINARYSORT(:1)
```

Similar constructs can be used with the BETWEEN predicate.

Note. The %BINARYSORT meta-SQL token ensures that the database evaluates the column that is wrapped by the token based on its binary value in the character set of the database. The sorting of this representation matches its binary sort position in PeopleTools only if the character set of the database contains the character in the same order as Unicode, which is used for binary representation of data in PeopleTools memory. Therefore, if you are running a US-ASCII, Latin-1 (ISO 8859-1), or Unicode database, the sorting of all alphabetic characters matches the Unicode sort in memory. However, if you are running a database that is encoded in EBCDIC or in Japanese Shift-JIS, the order may be markedly different, and you cannot rely on the %BINARYSORT meta-SQL token to match the binary order that is used within PeopleTools.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleCode Language Reference, “Meta-SQL,” %BINARYSORT

Sorting in COBOL

Character string comparisons that are performed in COBOL are performed based on the binary representation of each character in the character set that is being used by the COBOL program. Typically, this matches the character set of the database. Use the %BINARYSORT meta-SQL token to wrap any ORDER BY clause or string comparison that is performed by the SQL statements that are called by the program that you want to sort the same as it would sort in COBOL memory.

Sorting in SQR

Like PeopleCode, all string comparisons and sorting in SQR are performed based on a Unicode binary sort. However, because SQR does not process PeopleTools meta-SQL tokens, you cannot use %BINARYSORT from SQR. You should make allowances for SQR programs to not depend on having memory sorting match the sort that the database performs in ORDER BY and similar SQL clauses.

CHAPTER 8

Understanding Global Reporting and Data Analysis Tools

This chapter discusses:

- Language-sensitive queries.
- PeopleSoft Cube Manager for global implementations.
- The Strings table and language-sensitive text in reports.
- SQR for PeopleSoft globalization.
- PS/nVision reporting for global implementations.
- International versions of Microsoft Excel.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Crystal Reports for PeopleSoft, “Using Crystal Reports,” Working with Multiple Languages

Understanding Language-Sensitive Queries

PeopleSoft Query supports language-sensitive query output. If you create a query on a table that has a related language record, or if your query includes such a table, PeopleTools automatically performs the language lookup on the related language record. This means that the output of the query appears in the current, preferred language, if translations exist in the related language tables.

For example, if you built a simple query on the Country table (COUNTRY_TBL), the country descriptions in your query would appear in your preferred language, even though you queried the base language record (COUNTRY_TBL) and not the related language record (COUNTRYTBL_LANG). PeopleSoft Query performs the necessary join to the related language table to retrieve the translations; if they don't exist, it retrieves the descriptions in the database's base language.

The related language join is performed only when you run the query. Therefore, if you create a query as an English user and then execute the same query as a French user, the output appears in French, even though the query was created in English. The language preference of the user who is running the query drives the language joins.

Note. WHERE criteria, HAVING criteria, and ORDER BY clauses are applied only to the base-language data. For example, assuming that the base language of your database is English, if you set a WHERE criterion that restricts country descriptions to those that begin with *Ger*, PeopleSoft Query retrieves the *Allemagne* country description when the query is run by a French language user. This is because the English description of *Allemagne* is *Germany*.

The automatic language join features in PeopleSoft Query are also used when you access your PeopleSoft database via the PeopleSoft Open Query ODBC interface.

Scheduled Queries

For scheduled queries, the system uses the language specified in the user's profile. It does not use the language selected during sign-in. The system will also use the international and regional settings the user has specified using the My Personalizations page. If no personal setting have been specified, the system uses the default installation international settings.

Note. Most PeopleSoft components can default to international settings from the browser if the user has not set any user-specific settings. However, this is not available for scheduled queries or any Process Scheduler processes.

See Also

PeopleTools 8.45 PeopleBook: PeopleSoft Query

Understanding PeopleSoft Cube Manager for Global Implementations

In PeopleSoft Cube Manager, you can build Cognos PowerPlay cubes and Hyperion Essbase cubes that display translated data. The cubes themselves do not have to be translated; they work by accessing the translations (stored in related language tables) of language-sensitive query output and tree nodes that correspond to the cube's members.

This section discusses:

- Cognos PowerPlay local-language cubes.
- Hyperion Essbase multiple-language cubes.

See Also

[Chapter 15, "Translating Application Definitions," Translating Trees, page 238](#)

[Chapter 11, "Working With Language-Sensitive Application Data," page 169](#)

PeopleTools 8.45 PeopleBook: PeopleSoft Cube Manager

Cognos PowerPlay Local-Language Cubes

Cognos PowerPlay cubes are not language-sensitive. You must build a separate cube in each language that your implementation supports. Users must then access the language version of the cube that matches their language preferences.

When you build a PowerPlay cube instance, you can override the default language in which the cube is built. If you do not override the language, PeopleSoft Cube Manager builds the PowerPlay cube using the database's base language.

Each PowerPlay cube instance is built using a single language. This language can be the database's base language or any non-base language that your implementation supports. If the cube is built in the base language, then all language-sensitive elements appear in the base language. If the cube is built in a non-base language, then the cube's language-sensitive elements appear in that language using translations of language-sensitive query output and tree nodes. If the language-sensitive fields for the query output and tree nodes have not been translated into the cube's language, the cube displays the base-language descriptions.

To override the language of a Cognos PowerPlay cube:

1. Select PeopleTools, Cube Manager, Build, Cube Instances.

Use the standard search method to access the Cube Inst Def (cube instance definition) page.

2. Confirm that the Analysis Platform field is set to *Cognos PowerPlay*.
3. Click the Platform Options link.

The PowerPlay Options page appears.

4. Select the desired language in the Language Override field.
5. Click OK to close the page.
6. Save the component.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Cube Manager, "Designing Cubes"

Hyperion Essbase Multiple-Language Cubes

When building a Hyperion Essbase cube instance, you specify the languages that the cube supports. These can be the database base languages and any other languages supported by the implementation. When users access the cube in Essbase Administrator (or in the Essbase Excel Client), they can choose to view aliases for the cube members (stored on Essbase alias tables) in any of the supported languages. The translations of the aliases are derived from language-sensitive query output and tree nodes, so the translated aliases appear only if the tree nodes have been translated and the query data is language-sensitive and has been translated. Specific elements that have not been translated appear in the database's base language.

See the Essbase documentation for information about using Essbase alias tables.

Specify the languages that supported by an Essbase cube when you define the Essbase cube instance. If no languages are specified, no local-language Essbase alias tables are built, and the cube member descriptions appear in the database's base language.

To specify the languages of an Essbase cube instance:

1. Select PeopleTools, Cube Manager, Build, Cube Instance.

Use the standard search method to access the Cube Inst Def (cube instance definition) page.

2. Confirm that the Analysis Platform field value is *Hyperion Essbase*.
3. Click the Platform Options link.

The Essbase Advanced Options page appears.

4. From the Language Code dropdown list, select the cube's supported languages.

To set the value of a prompt field, click in the field and then choose from the list of supported languages. To insert a new language, click the Add button. To remove a language from the list of languages that are supported by the cube, click the Delete button.

5. Click OK to close the page.
6. Save the component.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Cube Manager, “Designing Cubes”

Understanding the Strings Table and Language-Sensitive Text in Reports

The PeopleTools Strings table (STRINGS_TBL) stores textual strings used for language-sensitive labels and other text in PS/nVision and SQRs to avoid hard-coding labels into the report files themselves. The use of strings rather than hard-coded text in reports enables translators to translate the report layout in the database without editing the report’s code itself. This enables you to run a single copy of a report in multiple languages, while avoiding the duplication of code and report logic.

Each row in the Strings table keys each string to a STRING_ID, which is associated with one of two different string types:

- The short (RFT Short) or long (RFT Long) field description of the field.

The STRING_ID must equal the field name.

- A free-form text string.

In either case, when a report requests a string in a particular language, the system returns the translated string if a translation is available. If no translation exists, the base language string is returned.

The Strings table is also keyed by a PROGRAM_ID field, which enables you to classify strings into groups that are used in similar reports. PROGRAM_ID can refer to a specific SQR or PS/nVision report name, or it can be a mnemonic for a group of common strings that are shared between reports.

Num	Field Name	Type	Key	Ord	Dir	Cur	Srch	List	Sys	Audt	Defa
1	PROGRAM_ID	Char	Key	1	Asc		Yes	Yes	No		
2	STRING_ID	Char	Key	2	Asc		No	Yes	No		
3	LABEL_ID	Char					No	No	No		
4	DEFAULT_LABEL	Nbr					No	No	No		
5	STR_LBLTYPE	Char					No	No	No		'1'
6	STRING_TEXT	Char					No	No	No		

STRINGS_TBL record definition

See Also

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” Understanding SQR for PeopleSoft Globalization, page 107](#)

[Chapter 15, “Translating Application Definitions,” Translating Strings, page 242](#)

Evaluation of PS/nVision String Variables

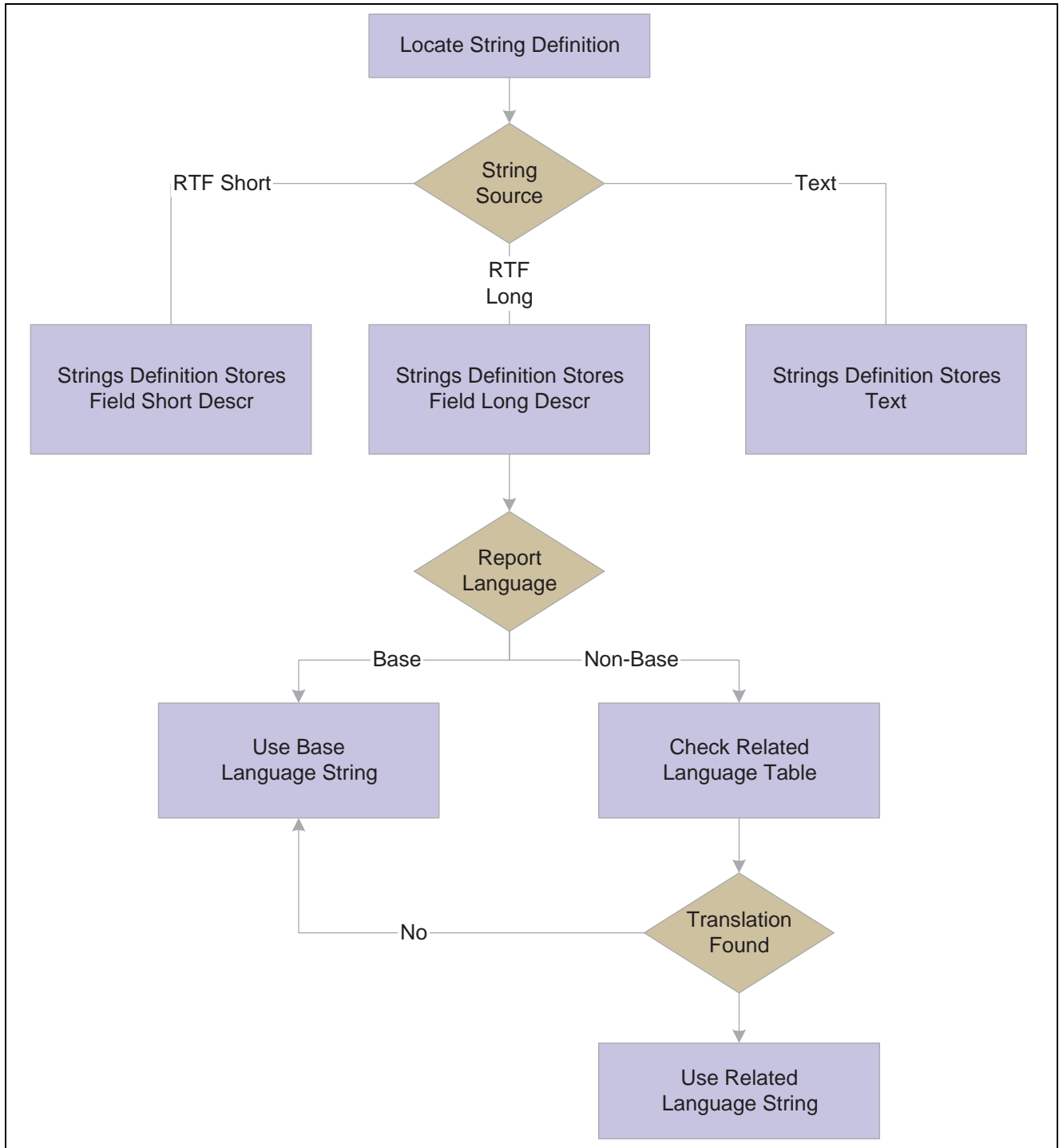
The system uses the string definitions in the Strings table to evaluate PS/nVision variables based on three factors:

- A program ID, which identifies a set of strings used in similar reports.
- A string ID, which uniquely identifies the string definition within a program ID group.
- The language of the report request. If no language is specified, the system assumes the preferred language of the current user.

The syntax of a PS/nVision string variable includes the string ID and the program ID:

```
%.StringID,ProgramID%
```

The following flowchart shows how the system evaluates language-sensitive string variables:



Evaluation of SQR and PS/nVision string variables

The evaluation follows this sequence:

1. The system locates a string definition in the Strings table using values for the program ID and string ID.
The text string stored in the Strings table (or its related language table) can be a short or long field description (validated against the PSDBFIELD or PSDBFIELDLANG table) or a free-form text string that is not associated with a field.

2. The system determines the language of the report request. If the value is unspecified, the system uses the user's current language preference setting.
3. If the language is the system's base language, the system retrieves the string values from the base-language Strings table (STRINGS_TBL).
4. If the system is a non-base language, the system looks for a string with the correct language code in the related-language table (STRINGS_LNG_TBL).
5. If no translation exists in the related-language table, the system uses the string in the base-language table (Default).

See Also

[Chapter 15, "Translating Application Definitions," Translating Strings, page 242](#)

Understanding SQR for PeopleSoft Globalization

SQR for PeopleSoft provides a range of features to allow not only for multilanguage reports, but also for handling international date/time formatting, paper sizes, numeric formatting and much more. This section discusses some of the internationalization features of SQR for PeopleSoft:

- Printing for A4 paper.
- Currency precision.
- Date and time formatting.
- Report translation
- PSSQR.INI and PSSQR.UNX files.
- SQR configuration for processing international text.
- SQR configuration for printing international text.
- International text in SQR for PeopleSoft programs.
- SQR for PeopleSoft supported character set encodings.

See Also

Enterprise PeopleTools 8.45 PeopleBook: SQR Language Reference for PeopleSoft, "SQR Command Reference"

Printing for A4 Paper

PeopleTools supports printing A4, Legal, and US Letter sized paper. Changing the PAPER_SIZE setting in SETENV.SQC changes the paper size for all reports that share the same file/report/batch server directory.

To print Legal paper, uncomment the following line in the SETENV.SQC file:

```
#define PAPER_SIZE LEGAL
```

To print A4 paper, uncomment the line above and edit it to be:

```
#define PAPER_SIZE A4
```

To ensure that you are printing your desired paper size, you must comment out or delete any other #define PAPER_SIZE statements.

Currency Precision

NUMBER.SQC enables you to make use of currency precision, both in terms of character string values (with an edit mask) and numeric values, rounded to a specified precision. Use these functions to achieve your currency precision requirements:

```
Format_Currency_Amt
Format_Currency_Amt_Numeric
```

See the comments block at the beginning of the NUMBER.SQC SQR include file for documentation of these functions.

Date and Time Formatting

The DATETIME.SQC program provides several procedures to aid in the formatting of date and time values for display in report output. While DATETIME.SQC provides generic procedures for formatting dates and times, the actual format used for date and time values can vary in each report. By default, when printing dates and times, reports use the system-wide default date and time formats that are specified in SETENV.SQC.

During your PeopleSoft installation, you should edit SETENV.SQC to specify the system-wide default format you prefer for date and time values. Edit the following lines in your SETENV.SQC to tell the SQR which format you prefer, if it is not otherwise specified in the SQR report source.

```
#define Year4      '1'          !0 = 2 digit year
#define Prompt-Date mm/dd/yyyy
#define Prompt-Mask 'MDY'
#define DateType  '0'          !iDate 0 = mdy, 1 = dmy 2 = ymd
#define TimeDisplay '1'        !iTime 0 = 12hr, 1 = 24hr
```

Based on these settings or any overrides, procedures in DATETIME.SQC provide support for various date formats. The following table identifies these formats.

SQR Date Format	Description
{DEFDMY}	DD/MM/YYYY
{DEFMDY}	MM/DD/YYYY
{DEFKAN}	Japanese Kanji-format dates (using the Japanese Imperial calendar).
{DEFROM}	Japanese Romaji-format dates (using the Japanese Imperial calendar).

The following table includes examples for SQR for PeopleSoft date formats for the date December 14, 2000.

14/12/2000 {DEFDMY}

12/14/2000 {DEFMDY}
 + {DEFKAN}
 H.12.12.14 {DEFROM}

For details on how to include date/time formatting in your own SQR reports, refer to the documentation inside the comment block of the DATETIME.SQC SQR include file on your file server. If you plan to use Japanese date formatting in your SQR report, remember to include *#define JapaneseDates* at the top of your SQR report.

Report Translation

PeopleTools enables you to print or format SQR output for multiple languages using string definitions that are stored in the Strings table. The procedures defined in SQRTRANS.SQC enable your SQR program to access these strings dynamically.

To enable the use of the Strings table in your SQR, you must include the PeopleTools SQC file SQRTRANS.SQC, which includes the routines that are necessary to initialize and load translated strings from the Strings table.

SQRTRANS.SQC has four main functions that you can call from the report:

- Init_Report_Translation
- Get_Field_Information
- Append_Report_Translation
- Add_Report_Translation

Init_Report_Translation

Call the Init_Report_Translation procedure from your SQR before using any of the String table information. Typically, you should call Init_Report_Translation in the Init-Report section of your SQR. Init_Report_Translation takes two parameters:

Parameter Name	Description
\$Report_ID	\$Report_ID is normally the name of your SQR. This parameter is used as the program ID when looking up strings in the Strings table.
\$Report_Language	\$Report_Language indicates the preferred language for the strings that are being retrieved. Init_Report_Translation attempts to load all strings in the language specified; however, if a translation for any string does not exist, it loads the base language description for that string.

If you want to change languages during the processing of an SQR report (for example, if you want each page of the report to be in a different language), you can call Init_Report_Translation multiple times within a single SQR, each time passing a new \$Report_Language value.

Get_Field_Information

Call the `Get_Field_Information` procedure for each string that you want to retrieve from the Strings table. It retrieves the label or string table entry for the field specified and places it in a report variable. You can then print the contents of this variable on your report as a label, column heading, or free text. `Get_Field_Information` takes four parameters:

Parameter Name	Description
<code>\$Report_ID</code>	<code>\$Report_ID</code> is normally the name of your SQR. This parameter is used as the program ID when looking up strings in the Strings table. You must have already called <code>Init_Report_Translation</code> specifying this <code>\$Report_ID</code> before passing it to <code>Get_Field_Information</code> .
<code>\$Field_ID</code>	This is the string ID of the string whose text you want to retrieve. It must exist as an entry in the Strings table under the <code>\$Report_ID</code> that you specified.
<code>\$Field_Text</code>	<code>\$Field_Text</code> is the output variable. <code>Get_Field_Information</code> populates this variable with the text that corresponds to the <code>\$Report_ID</code> and <code>\$Field_ID</code> that are specified in the preferred language or in the database's base language (if a translation doesn't exist in the preferred language).
<code>\$Field_Width</code>	<code>\$Field_Width</code> is an output variable that <code>Get_Field_Information</code> populates with the width of the text string that is returned.

Append_Report_Translation

If your SQR program uses strings from more than one Strings table program ID, call `Append_Report_Translation` to add the strings from another program ID to the initialization array created by `Init_Report_Translation`. This function is particularly useful if you have a set of strings that are used across many of your SQR programs. You can group these strings under a generic program ID and use them in multiple SQRs.

`Append_Report_Translation` takes a single argument, `$Report_ID`. It assumes the same language that was specified in `Init_Report_Translation`. It must be called after `Init_Report_Translation`.

Add_Report_Translation

The `Add_Report_Translation` procedure calls `Init_Report_Translation` or `Append_Report_Translation`, depending on whether the Strings table has been initialized. It takes the same arguments as `Init_Report_Translation`. If `Init_Report_Translation` has not yet been called during the processing of this SQR, this function calls it, passing both parameters. If `Init_Report_Translation` has already been called, `Add_Report_Translation` calls `Append_Report_Translation`, passing only the `$Report_ID` parameter.

This function is useful in your own SQC files if you cannot be certain that the calling SQR program has already initialized the Strings table. The function ensures that the table is initialized or appended correctly.

Sample Strings Table Enabled SQR

The following sample code demonstrates how to use the Strings table to retrieve string values in SQR, using the procedures described in the preceding sections.

```

!*****
! SAMP001: Report on database's base language *
!*****
#include 'setenv.sqc'      !Set environment
begin-report
  do Init-Report
  do Process-Main
  do Reset
end-report
begin-heading 6
do Get_Field_Information ('SAMP001',      'REPORT_TITLE', $REPORTTITLE,  #DWRT)
do Get_Field_Information ('SAMP001',      'EXPLAIN_TEXT', $EXPTEXT,      #DWET)
PRINT $REPORTTITLE      (1)      CENTER
PRINT $EXPTEXT          (+2,1)
end-heading
begin-procedure Init-Report
move 'SAMP001' to $ReportID
move 'ENG' to $Language_cd
do Init_Report_Translation ($ReportID, $Language_cd)
do Append_Report_Translation ('GEN')
end-procedure
begin-procedure Process-Main
do Get_Field_Information ('GEN',          'BASELANGUAGE', $BASELANGUAGE, #DWBL)
do Get_Field_Information ('GEN',          'ENDOFREPORT', $ENDOFREPORT,  #DWER)
begin-SELECT
LANGUAGE_CD &Base_Language
  let $langlabel = $BASELANGUAGE || ':'
  print $langlabel (+1,1)
  let #fieldpos = #DWBL + 3
  print &Base_Language (0,#fieldpos)
FROM PSOPTIONS
end-SELECT
print $ENDOFREPORT      (+4,1)
end-procedure
#include 'reset.sqc'      !Reset printer procedure
#include 'sqrtrans.sqc'  !SQR Strings Table procedures

```

See Also

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” Understanding the Strings Table and Language-Sensitive Text in Reports, page 104](#)

The PSSQR.INI and PSSQR.UNX Files

Most of the parameters that affect globalization of SQR for PeopleSoft are set in the configuration file. PeopleSoft delivers default configuration files, which are located in the %PS_HOME%/sqr directory. The files are named pssqr.ini on Windows, and pssqr.unx on UNIX/Linux. Because different configurations are needed for some of the supported languages, PeopleSoft delivers eight configuration files on each platform. These language-specific configuration files are named pssqr<language code>.ini or pssqr<language code>.unx. When the SQR process is started from PeopleSoft Process Scheduler, one of these configuration file is selected based on user's language preference set in the My Personalization page. When language-specific files are not found, Process Scheduler selects pssqr.ini/unx without <language code>.

By default, these configuration files are assigned a character set and font that are correct for each language. However, it is also possible to customize the files to better suit your integration needs and report requirements. The following sections describe the details of configurations you can specify in the pssqr.ini/unx files.

Note. On DB2 UDB for OS/390 and z/OS initialization files, PSSQR<LANGUAGECD>INI and PSSQRINI are provided as members of the SQRSRC dataset on TSO. Process Scheduler selects a language-specific initialization file if it exists in the SQRSRC dataset by replacing the %SQRINI% meta-string on shell JCT at process run-time. Because of the limitation in Process Scheduler running on z/OS, it is not possible to use a different character set on language-specific INI files on z/OS. The language-specific configuration files are provided to make different font configurations for some languages. If you need to process multiple character sets for a single database using Process Scheduler on z/OS, you must set up multiple Process Schedulers.

See Also

[Chapter 8, "Understanding Global Reporting and Data Analysis Tools," SQR Configuration for Processing International Text, page 112](#)

[Chapter 8, "Understanding Global Reporting and Data Analysis Tools," SQR Configuration for Printing International Text, page 113](#)

SQR Configuration for Processing International Text

SQR for PeopleSoft uses Unicode for internal storage of character data regardless of whether you are running against a Unicode encoded PeopleSoft database. This enables a single instance of SQR to process data in virtually any language.

Although SQR runs using Unicode internally, it automatically converts data read from and written to files to the appropriate non-Unicode encoding based on configurations made in the pssqr.ini/unx file. You can modify the delivered configuration to use different non-Unicode encoding based on your integration and reporting needs. Further, you can use Unicode as the input and output character set if your printer or external system communicating with SQR by file can process Unicode data.

PeopleSoft delivers pssqr.ini/unx files with configurations for each supported language and database connection. Manual configuration is not required for basic processing of supported languages or to connect to a Unicode database. The information in this section is provided to aid you in customizing SQR for PeopleSoft to better suit your integration and reporting requirements.

See [Chapter 8, "Understanding Global Reporting and Data Analysis Tools," The PSSQR.INI and PSSQR.UNX Files, page 112](#).

This section describes advanced pssqr.ini/unx encoding settings.

Advanced PSSQR.INI/UNIX Settings

The following parameters under the [Environment: Common] or [Environment: <Database type>] sections of the pssqr.ini/unx file control the character set SQR uses for specific operations. The ENCODING parameter sets the default encoding for all types of operations. You can, however, control encodings for each operation independently by adding the parameters in the [Environment: Common] section of the file.

```
ENCODING
ENCODING-SQR-SOURCE
ENCODING-FILE-OUTPUT
ENCODING-FILE-INPUT
ENCODING-REPORT-OUTPUT
ENCODING-CONSOLE
```

If an individual encoding parameter is explicitly included, this specific setting overrides the default encoding specified by the ENCODING parameter.

- The ENCODING-SQR-SOURCE parameter specifies the encoding in which the SQR source (*.SQR, *.SQC) files are encoded.
- The ENCODING-FILE-OUTPUT and ENCODING-FILE-INPUT parameters control the character set that is used to read and write files using the SQR OPEN command. If no character set is specified explicitly in the OPEN command, the character set specified in these parameters is used to read or write the file. If SQR detects a BOM at the top of file, it always reads as a Unicode file, regardless of this setting.
- The ENCODING-REPORT-OUTPUT parameter determines the encoding used for report output types LP, HP, PS, and CSV. The parameter should be set to a character set that contains all the characters you expect to print using SQR and, if you intend to print the output on a printer, it should also be set to the character set that the printer supports. Output in SPF, HTML, Enhanced HTML, and PDF format is not controlled by this parameter. SPF HTML, and Enhanced HTML always use Unicode for their output, and PDF output is controlled exclusively by the font configuration.
- The ENCODING-CONSOLE parameter determines the character set that is used to write progress and other messages to the console during an SQR run. In Windows, the encoding used by DOS consoles, such as the one that is used by SQR, is known as an OEM encoding; this is often different from the character set that is used by Windows. See <http://www.microsoft.com/globaldev> for a list of Microsoft Windows OEM codepages. For UNIX/Linux systems, this setting should match the character set that is supported by your terminal device.
- If the ENCODING parameter is not set, SQR determines the encoding to use based on the machine locale.
- If the same parameter is not set in the [Environment: Common] and [Environment: <Database Type>] sections of the pssqr.ini/unx file, the configuration in the [Environment: <Database Type>] section overrides the configuration in the [Environment: Common] section.

See Also

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” The PSSQR.INI and PSSQR.UNX Files, page 112](#)

SQR Configuration for Printing International Text

SQR for PeopleSoft is a Unicode application. It can generate reports that contain multiple languages, such as English, French, Japanese, and Thai, on a single report and even on a single page, if you use advanced reporting output types like PDF, SPF, HTML, or Enhanced HTML. However, for other output types SQR supports, printing or generating reports in languages other than the Western European languages might require special considerations. This section provides some hints for customizing or resolving international text printing issues with SQR for PeopleSoft.

This section discusses how to:

- Set up encoding parameters.
- Set up fonts.
- Set up PDF fonts.
- Set up files for PCL and line printing.
- Set up files for PostScript printing.

This section also discusses limitations of SQR for PeopleSoft to print international text.

Setting Up Fonts

SQR for PeopleSoft represents fonts internally as numbers, and the pssqr.ini/unx file controls the mapping between the font numbers for some printer types. The [Fonts] section in the pssqr.ini/unx file controls the font mappings of the following output types:

- SPF Viewer
- WP (Windows Printer)
- Enhanced HTML

By default, this section is not configured with language-specific fonts. This is because Windows and HTML browsers have a font linking mechanism that allows them to select an appropriate language-specific font automatically when the font specified does not have characters to show. However, Windows and the browser's font linking works differently based on versions, availability of fonts on the system, and the configuration of Windows and the browser. If you experience problems showing characters for these output types, you should try the configuration described in this section

To set up fonts:

1. Open the appropriate pssqr.ini/unx file for your language, located in %PS_HOME%/SQR.
2. In the [Fonts] section, the following default values display. Use the chart to replace each of the entries for your target language.

```

3=Courier New,fixed
300=Courier New,fixed,bold
4=Arial,proportional
400=Arial,proportional,bold
5=Times New Roman,proportional
500=Times New Roman,proportional,bold
6=AvantGarde,proportional
8=Palatino,proportional
800=Palatino,proportional,bold
11=Symbol,symbol
900=unknown,proportional
901=Times New Roman,proportional
28825=MS UI Gothic,proportional

```

Font Number	Japanese	Simplified Chinese	Traditional Chinese	Thai	Korean
3	MS Gothic, proportional	SimHei, proportional	MingLiu, proportional	CordiaUPC, proportional	GulimChe, proportional
300	MS Gothic, proportional, bold	SimHei, proportional, bold	MingLiu, proportional, bold	CordiaUPC, proportional, bold	GulimChe, proportional, bold
4	MS Gothic, proportional	SimHei, proportional	MingLiu, proportional	CordiaUPC, proportional	GulimChe, proportional
400	MS Gothic, proportional, bold	SimHei, proportional, bold	MingLiu, proportional, bold	CordiaUPC, proportional, bold	GulimChe, proportional, bold
5	MS Mincho, proportional	SimSun, proportional	MingLiu, proportional	AngsanaUPC, proportional	BatangChe, proportional
500	MS Mincho, proportional, bold	SimSun, proportional, bold	MingLiu, proportional, bold	AngsanaUPC, proportional, bold	BatangChe, proportional, bold
6	MS Gothic, proportional	SimHei, proportional	MingLiu, proportional	CordiaUPC, proportional	GulimChe, proportional
8	MS Gothic, proportional	SimHei, proportional	MingLiu, proportional	CordiaUPC, proportional	GulimChe, proportional
800	MS Gothic, proportional, bold	SimHei, proportional, bold	MingLiu, proportional, bold	CordiaUPC, proportional, bold	GulimChe, proportional, bold
901	MS Mincho, proportional	SimSun, proportional	MingLiu, proportional	CordiaUPC, proportional	GulimChe, proportional
28825	MS UI Gothic, proportional	N/A	N/A	N/A	N/A

Setting Up PDF Fonts

For PDF output, SQR for PeopleSoft has an internal font linking mechanism by which you can link up to 10 fonts for a single font number. This allows you to get character coverage that is not possible by using single font. With this feature, you can print reports containing multiple languages without rewriting your program to show each language in a different font. PeopleSoft delivers the `pssqr.ini/unx` file with world-ready configuration; therefore, you usually do not need to modify the `pssqr.ini/unx` file to print non-Western European languages. Many major business languages can be printed in PDF independent of the user's language settings on the My Personalization page. For example, you can output Greek data in PDF even when your My Personalizations language is set to English.

Although it is possible to print Chinese, Japanese, or Korean text in a PDF report without modifying the `pssqr.ini/unx` configuration, printing Chinese, Japanese, and Korean text in a single report requires special configuration. Font linking data is configured in the `pssqr.ini/unx` file and is transparent to the user. Depending on your reporting requirements, you can add or remove fonts from the delivered configuration, and you can control which actual font to use down to a single character level. This section explains how to add fonts for PDF output.

The [PDF Fonts] section in the `pssqr.ini/unx` file supports multiple fonts mapped to a single font number.

The [PDF Fonts: Exclusion Ranges] section of the file specifies exclusion ranges for fonts listed in the [PDF Fonts] section. If an exclusion range is set, then when a character is covered by a font, that font is not used to print the character. Exclusion ranges are set in Unicode format, in hexadecimal or decimal. The base configuration does not contain an exclusion range.

The [TrueType Fonts] section of the file specifies the mapping from TrueType font names used in the [PDF Fonts] configuration section, along with the physical file path of the font on the operating system. For TrueType collection (.ttc) files, the font directory number should also be specified (in the format *font name=file path, directory number*).

The font path is the directory where the font resides. The default is `SQRDIR`. On Windows, the font directory is looked up as well. Fonts residing in other directories must be specified by their full physical pathnames.

The [PDF Fonts] information in the `pssqr.ini/unx` file is language-specific for Chinese, Japanese, and Korean languages; the information is the same for other supported languages due to Unicode characteristics. If you want to add another language (for example, Russian), you should refer to the documentation for adding new languages.

See PeopleTools 8.45 Installation Guide for Your Database Platform

See [Chapter 13, "Adding New Languages," page 199](#).

See *Enterprise PeopleTools 8.45 PeopleBook: SQR Language Reference for PeopleSoft*.

Sample Steps for Adding a Font for PDF Output: User Defined Font for Japanese

For computing in the Japanese language, it is common to use user-defined characters. A user-defined character is a character that is not encoded in standard character sets like JIS X 0208 or Unicode, and sometimes variants of glyphs existing in standard character sets. This kind of character is often used for a person's name. Unicode has space allocated to encode such user-defined characters, and users can create fonts that include characters not defined in Japanese fonts.

On Windows, users can create user-defined fonts using a program called EUDCEDIT to design and define private characters. This program creates a TrueType font file named "eudc.tte" under the Windows font folder. As an example of adding fonts for PDF output from SQR for PeopleSoft, perform the following steps to add the "eudc.tte" font to the PeopleSoft-delivered configuration file for Japanese, `pssqrjpn.ini`:

1. Open the pssqrjpn.ini file from %PS_HOME%/sqr and add the following line under [TrueType Fonts] section:

```
[TrueType Fonts]
Font Path=...
Cumberland=...
GaijiFont=eudc.tte
```

As explained in this section, you do not need to specify a full path for the font residing in the Windows font folder, and you can specify a name different from the actual name of the font ("GaijiFont" means a user defined font in Japanese. The actual font name that "eudc.tte" itself has is "EUDC").

2. Under the [PDF Fonts] section, add "GaijiFont" at the end of the font list. For example, for font number 3:

```
3=Courier,HeiseiKakuGo-W5,Cumberland,MHei-Medium,STSong-Light,HYGoThic-Medium,
Angsana,GaijiFont
```

3. Define exclusion ranges in the [PDF Fonts: Exclusion Ranges] section, to exclude higher priority fonts to show characters in the Unicode private use range where characters in eudc.tte are defined. Chinese fonts have glyphs defined in the Unicode private use range mainly to include characters defined in national character standards like GBK but not defined in earlier versions of Unicode. Thai fonts also have Thai presentation form characters mapped in the Unicode private use range. As we are working on a Japanese configuration and Chinese characters and Thai presentation forms are rarely needed in a Japanese context, we will configure these fonts not to print characters in the Unicode private range, and private range characters are always printed using "GaijiFont" we have just added. The Unicode private use range is from 0xE000 to 0xF8FF. Chinese fonts are third and fourth in the font link list, and the Thai font is seventh. Therefore, the exclusion definition looks like the following (exclusion range setting for second font comes from original configuration of pssqrjpn.ini, that excludes extended Latin characters from Japanese font):

```
3=,0x0000-0x02FF,,0xE000-0xF8FF,0xE000-0xF800,0xE000-0xF800,,
```

Considerations for Printing Chinese, Japanese, and Korean Text in a Single PDF Output

Unicode application printing of Chinese, Japanese, and Korean text in a single context requires special consideration in the use of fonts, because Unicode encodes ideographs used in Chinese, Japanese, and Korean languages in unified codepoints. A Unicode character is unique in meaning, but the glyph (the shape of the character) might be different depending on the language context. A commonly used technique in resolving this glyph difference is to use a language-specific font for Chinese, Japanese, and Korean, based on external context.

PeopleSoft delivers different font linking configurations for Chinese, Japanese, and Korean languages in language-specific pssqr.ini/unx files. In the Simplified Chinese configuration, the Simplified Chinese font is given higher priority than other Chinese, Japanese, or Korean fonts, so that a Simplified Chinese glyph is used for most Chinese ideographs. At runtime, the Process Scheduler picks up language-specific pssqr.ini/unx files with appropriate font linking data for the language, based on the user's My Personalization language setting. If you set your language as Traditional Chinese, the configuration file that uses the Traditional Chinese font for the majority of ideographic characters will be picked up. PeopleSoft recommends that you set up the language in the My Personalization page when you are printing reports that include Chinese, Japanese, and Korean data. If you do not use any Chinese, Japanese, and Korean languages for My Personalization page, the default configuration is used and in this case the printed glyph image might be different from what you expect to see.

If you have a reporting requirement to print a mix of Chinese and Japanese in a single report, and you need to show each language in the correct glyph image for that language, you will need to use the ALTER-PRINTER command to programmatically change the font in your SQR program based on external data, such as the Language Code of the data. For this purpose, pssqr.ini/unx files are provided with the following font numbers, each of which uses fonts specific for each Chinese, Japanese, and Korean locale.

- 28825, 28752 (Japanese).

- 37110, 37058 (Simplified Chinese).
- 33269 (Traditional Chinese).

Setting Up Files for PCL and Line Printing

Most of the configurations for printing international text from SQR for PeopleSoft are performed by setting options in the pssqr.ini/unx file. However, for some of the printer-specific settings, you must set parameters in SQR for PeopleSoft programs because of the SQR for PeopleSoft syntax.

For example, SQR for PeopleSoft enables users to control printer parameters for PCL format in DEFINE-PRINTER or ALTER-PRINTER commands. PeopleTools provides default setup values for PCL format in three SQC files (ptset01.sqc, ptset02.sqc and ptset02a.sqc). If you intend to print languages in PCL format other than those covered by Latin1, you must add printer setup parameters to the SQC files.

Before you perform the configuration tasks described in this section, make sure that you have properly configured your printers to print in the target language.

If your printer does not have the fonts for the languages you want to print, you might need to purchase fonts in ROM from your printer vendor and install them on your printer.

The parameters you set for PCL printing must be consistent with the ENCODING parameters set in the pssqr.ini/unx file. You must set up the ENCODING parameters for each language before you set the parameters for PCL printing. See Setting Up Encoding Parameters in this chapter.

To set up PCL printing parameters for Japanese:

1. Open setenv.sqc file under %PS_HOME%\SQR.
2. Uncomment the following parameter:

```
#define PRINT_JAPANESE
```

3. Save and close the file.

To set up PCL printing parameters for languages other than Japanese:

Note. If you intend to print a language other than those covered by Latin1, and if it is not Japanese, follow these steps to setup the SQC files. The values presented here are examples and can vary depending on your printer. You should check your printer manual for the appropriate fonts and symbol sets.

1. Open the ptset01.sqc file located under %PS_HOME%\SQR.
2. Between the declare printer DEFAULT-HP and end-declare statements, add the following information with the appropriate values:

```
font
symbol-set
```

The values you set differ from language to language. The following table shows sample values for several languages commonly used by PeopleSoft customers.

Language	Font	Symbol Set
Central/Eastern European	No need to set.	2N

Language	Font	Symbol Set
Traditional Chinese	33269	18T
Simplified Chinese	37058	18C

For example, if you are setting ptset01.sqc for Simplified Chinese printing, try the following setting:

```
declare-printer DEFAULT-HP
#ifdef PRINT_JAPANESE
  init-string=<27>&t31P
  font=28825
  symbol-set=19K
#endif
font=37058
symbol-set=18C
point-size=7.2
pitch=17
end-declare
```

- Under the declare printer DEFAULT-LP statement, locate the init-string parameter.

Modify the init-string parameter based on the language to print. The following example shows sample values for HP LaserJet printer's line printer mode. Consult your printer manual for correct values for this parameter.

Language	init-string
Eastern European	<27>E<27>(2N<27>&10O<27>&16C<27>&10E<27>&195F<27> (s16.66H<27>&k2G
Traditional Chinese	<27>E<27>(18T<27>&10O<27>&16C<27>&10E<27>&195F<27> (s16.66H<27>&k2G<27>(s1p7.25v0s0b33269T
Simplified Chinese	<27>E<27>(18C<27>&10O<27>&16C<27>&10E<27>&195F<27> (s16.66H<27>&k2G<27>(s1p7.25v0s0b37058T

- Save and close the file.
- Open ptset02.sqc from %PS_HOME%\SQR and repeat steps 2 through 4.
- Open ptset02a.sqc from %PS_HOME%\SQR and repeat steps 2 through 4.

Setting up Files for PostScript Output

In addition to PCL and line printer format, SQR for PeopleSoft supports output in PostScript format, which can be printed with a printer that has PostScript interpreter. However, you configure fonts in a setup file provided for PostScript printing.

The parameters you set for PostScript printing must to be consistent with the ENCODING parameters you set in the pssqr.ini/unx file.

By default, the setup file provided for PostScript printing is configured for printing Western European languages. Depending on the encoding parameter in the pssqr.ini/unx file, you must either “re-encode” fonts, or use different fonts that support specific languages other than those set in the configuration file by default (you may need to use printer pre-installed fonts that support specific languages, or install external fonts on your printers). Consult your printer manual for the character sets your printer supports.

To set up files for PostScript output:

1. Open the postscri.str file from %PS_HOME%\bin\sqr\%DB Platform%\binw, or from the directory where the SQR executable resides.
2. If you are using select character sets such as Latin9, it is possible to print the entire repertoire of the character set just by “re-encoding” PostScript fonts. The PeopleSoft system delivers PostScript code and re-encoding data for Windows CP1252 and CP1250 on Microsoft Windows, and Latin1, Latin9, and Latin2 on UNIX/Linux. The default is CP1252 on Windows and Latin1 on UNIX/Linux. If you are using other character sets, the PeopleSoft system provides re-encoding data and you will need to modify part of the setup file. For example, the following example is for using the Latin9 character set on UNIX/Linux platforms:

Locate the following section:

```
% ISO Latin1
/COE [ ]
160 /space
...
```

Comment out the “CODE” section entirely, by adding a percent sign (%) at the beginning of each line.

Locate the following section:

```
% ISO Latin9
% (enable this array when you set ENCODING/ENCODING-REPORT-OUTPUT=ISO-8859-15
% and comment out above array for ISO Latin1)
% /COE [
% 160 /space
...
```

Remove the percent signs in front of each line for the entire “CODE” section.

Note. If you plan to use CP1250 on Microsoft Windows or Latin2 on UNIX/Linux, you will need to ensure that fonts installed on printer have the characters needed for these character sets. Depending on the printer and the fonts installed, you may not be able to print the entire repertoire of these Eastern European character sets using provided re-encoding data. Note also that if you plan to use character sets other than those discussed previously in this section, you will need to ensure that the printer has fonts supporting the specific character set you want to use. You may need to install additional fonts, and may also need to re-encode the fonts by adding your own font re-encoding data.

3. Some other character sets, including character sets commonly used for Japanese, Korean, and Chinese, can be printed by using PostScript fonts supporting the specific language. You can replace the font names in the setup file in the following way:

Locate the following section:

```
/Fonts [ % Array of point sizes and font names
```

Replace each font with the name of the appropriate font installed on your printer that can print the target language. For CJK languages (Chinese, Japanese, and Korean) PostScript printers accept the font names in the following format:

(Font Name)-(CMap name)

Font Name represents the name of the font installed on your printer that supports the language that you want to print. *CMap* is a file the PostScript printer used to find the glyph that corresponds to the codepoint passed to the printer. You need to use the *CMap* that matches the encoding specified in the ENCODING parameter in the pssqr.ini/unx file.

The following example shows setup values for each of the CJK languages. Supported font names and CMap files may vary depending on your printer. For details, refer to your printer documentation.

Language	Report Output Encoding	Serif Typeface	San Serif Typeface
Japanese	SJIS	Ryumin-Light-RKSJ-H	GothicBBB-Medium-RKSJ-H
Traditional Chinese	Big5	MSung-Light-B5-H	MHei-Medium-B5-H
Simplified Chinese	CP936	STSong-Light-GBK-EUC-H	STHeiti-Regular-GBK-EUC-H
Korean	CP949	HYSMyeongJo-Medium-KSCms-UHC-H	HYGoThic-Medium-KSCms-UHC-H

If you do not need to re-encode the fonts you have set, you must comment out the following section entirely by adding a percent sign (%) in front of each line. (For CJK character sets, you do not need to re-encode fonts; however, you might need to do so for other languages.)

```
%
% Re-encode all the fonts except for symbol/dingbats
% (do not re-encode CJK fonts - if you intend to use CJK fonts comment
% out the below section entirely)
%
/Courier /Courier CODE ReEncode
/Courier-Bold /Courier-Bold CODE ReEncode
/Helvetica /Helvetica CODE ReEncode
/Helvetica-Bold /Helvetica-Bold CODE ReEncode
...
```

4. Save and close the file.

Limitations in Printing International Text from SQR for PeopleSoft

The following limitations apply when printing international text from SQR for PeopleSoft:

- SQR does not support languages that require bi-directional text rendering, such as Arabic and Hebrew.
- The default configuration of report formats other than PDF, SPF, WP, HTML, and Enhanced HTML is to use non-Unicode character sets (ISO or commonly used national standard character sets on UNIX/Linux, and the Windows character set on Windows). The ability to print language data for LP, HP, PS, CSV, and HT output types is limited by the character set used for output. For example, you can print Western European languages like English, French, Spanish, and Portuguese using the Latin1 character set

(which is used in `pssqr.unx`) but you cannot add Polish data using Latin1, since the Polish language uses characters not encoded in the Latin1 character set. You can configure the `pssqr.ini/unx` files to generate output types other than HT in Unicode-based character sets like UCS-2 or UTF-8, but in these cases you need to ensure that your printer or viewer application supports Unicode natively.

- If you intend to print output using the LP, HP, or PS format, SQR for PeopleSoft's ability to print international characters is also limited by the fonts installed on the printers. For example, you can print Japanese on a PostScript printer if a Japanese font like Ryumin-Light is installed on the printer. However, if your printer does not natively provide font support, the printer output with international text may not be printed as intended. Unlike with PDF output, SQR for PeopleSoft currently does not support downloading host-based fonts, including TrueType fonts, to the printer as part of the print job. Printing from Adobe Reader or SPF Viewer is not affected by this limitation.

International Text in SQR for PeopleSoft Programs

This section discusses:

- System variables to check encoding settings.
- String length.
- OPEN command.

System Variables to Check Encoding Settings

You can reference the values set for the ENCODING parameters in `pssqr.ini/unx` from your SQR for PeopleSoft programs using the following system variables:

- *\$sqr-encoding*

Stores value set to the ENCODING parameter of `pssqr.ini/unx`. If no value is set for this parameter, SQR automatically determines the appropriate encoding for input and output based on the operating system locale and stores values in this variable.

- *\$sqr-encoding-database-api*

Stores value set to ENCODING-DATABASE-API parameter of `pssqr.ini/unx` file. Except for DB2 UDB for Linux, Unix, and Windows, the value for this parameter is automatically determined by the system and cannot be overridden by editing `pssqr.ini/unx`. In this case, this variable stores the encoding value automatically determined and used by system. For example, on an Oracle platform, this variable value is always *UTF-8*.

- *\$sqr-encoding-file-input*

Stores value set to ENCODING-FILE-INPUT parameter in `pssqr.ini/unx` file. If no value is set for this parameter, the input files are considered to be encoded in the encoding specified in the ENCODING parameter and this variable stores the same value as *\$sqr-encoding*.

- *\$sqr-encoding-file-output*

Stores value set to ENCODING-FILE-OUTPUT parameter in `pssqr.ini/unx` file. If no value is set for this parameter, the same encoding set in the ENCODING parameter is used for file output encoding and this variable stores the same value as *\$sqr-encoding*.

- *\$sqr-encoding-report-output*

Stores value set to ENCODING-REPORT-OUTPUT parameter in `pssqr.ini/unx` file. If no value is set for this parameter, the same encoding set in the ENCODING parameter is used for report output encoding and this variable stores the same value as *\$sqr-encoding*.

- *\$sqr-encoding-source*

Stores value set to `ENCODING-SQR-SOURCE` parameter in `pssqr.ini/unx` file. If no value is set for this parameter, SQR source files are considered to be encoded in the character set encodings specified in the `ENCODING` parameter and this variable stores the same value as `$sqr-encoding`.

- *\$sqr-encoding-console*

Stores value set to `ENCODING-CONSOLE` parameter in `pssqr.ini/unx` file. If no value is set for this parameter, the same encoding set in the `ENCODING` parameter is used for console output encoding and this variable stores the same value as `$sqr-encoding`.

Note. The system variables are read-only and cannot be overwritten at runtime. They can be used with functions and commands that take encodings as parameters, such as **substr()**, **length()**, and **open**.

String Length

When you work with strings in SQR, you must consider three different ways of measuring string length:

- The number of characters in the string.
- The number of print columns occupied by these characters.

For example, characters from the Latin alphabet normally require one print column; characters in Japanese often require two.

- The number of bytes used to store the character.

SQR uses the Unicode UCS-2 method of encoding characters, which means that every character occupies two bytes; however, your database may use a different encoding that requires a different number of bytes for each character.

SQR provides the following functions to help you manage string length according to each of these criteria.

Note. Although SQR uses Unicode internally, it can still read/write with non-Unicode (ANSI) databases and files. Refer to the `OPEN` command in the SQR documentation for details.

- **Length()** and **Substr()**.

Length() and **Substr()** deal with the number of characters in a string. As the PeopleSoft field lengths (as defined in PeopleSoft Application Designer) are character-based, the **Length()** and **Substr()** functions are useful for calculating the length of the string as it is stored in the database.

For example, the following code determines whether string `&abc` will fit into a database field that is 10 characters long. If the string won't fit into the field, the code truncates the string to use only the first 10 characters.

```
If length(&abc) > 10 then
&abc = substr(&abc,1,10)
End-if
```

- **Lengthp()** and **Substrp()**.

Lengthp() and **Substrp()** deal with the number of print columns required to print the character using a monospace (nonproportional) font.

For example, the following code determines whether string `&abc` will fit into a print area that is 10 columns wide. If the string won't fit into the print area, the code truncates the string to use only the first 10 columns of characters.

```
If lengthp(&abc) > 10 then
&abc = substrp(&abc,1,10)
```

```
End-if
```

- **Lengtht()** and **Substrt()**.

Lengtht() and **Substrt()** deal with the number of bytes that the string occupies in a specified character set. Typically, you would use **lengtht()** and **substrt()** if you are writing to a file in a specific character set, and you need to check or limit the byte length of the string in the output file, as would be required by most interface files.

For example, the following code determines whether string *&abc* will require more than 10 bytes in an output file. If the string is more than 10 bytes, the code truncates the string to use only the first 10 bytes worth of characters. The SQR system variable *\$sqr-encoding-file-output* is used to reference the *SQR.INI ENCODING-FILE-OUTPUT* variable, which determines the default character set of any file that is written to by the SQR OPEN command. You can substitute any valid PeopleSoft encoding for the *\$sqr-encoding-file-output* variable.

```
If lengtht(&abc, $sqr-encoding-file-output) >10 then
&abc = substrt(&abc, $sqr-encoding-file-output,1,10)
End-if
```

OPEN Command

The SQR OPEN command, used to read and write files from within SQRs, enables the report designer to specify the character set of the file being opened. You can specify a character set explicitly in the OPEN command. If you do not specify a character set, the SQR uses the character set specified in the *pssqr.ini/unx* parameter *ENCODING-FILE-OUTPUT* or *ENCODING-FILE-INPUT*, depending on whether you are opening the file for reading or writing.

To integrate with a third-party system, specify the character set for SQRs to match the target data. For example, a mainframe-based payroll system may expect an EBCDIC format file. Specifying the character set directly in the OPEN command enables the SQR program to be independent of the *pssqr.ini/unx* settings and enables the SQR to create the file directly in the character set expected by the target system (without the need to convert the output in a separate step).

As another example, if you are generating text output for a mail merge in Microsoft Word, you can specify UCS-2 or Unicode-Little-Endian to the *ENCODING* parameter of the OPEN statement, so that Word can import the Unicode file including international text.

See Also

[Chapter 4, “Selecting and Configuring Character Sets and Language Input and Output,” page 35](#)

Enterprise PeopleTools 8.45 PeopleBook: SQR Language Reference for PeopleSoft, “SQR Command Reference,” OPEN

SQR for PeopleSoft-Supported Character Set Encodings

This section lists the character set encodings that are supported by SQR for PeopleSoft 8.45. The character set encodings are organized by the following character sets:

- Arabic
- Baltic
- Celtic
- Chinese (Simplified)
- Chinese (Traditional)

- Cyrillic
- Gurmukhi
- Greek
- Hebrew
- Icelandic
- Japanese
- Korean
- Latin
- Latin (Canadian French)
- Latin (Central European)
- Latin (Southeastern European)
- Malayalam
- Nordic
- Slavic
- Symbol
- Thai
- Turkish
- Unicode
- Vietnamese

Explanation of the Following Tables

The Encoding Parameter column lists the values that you can specify in the ENCODING parameter in the pssqr.ini/unx file.

ENCODING-REPORT-OUTPUT applies to all output types.

In the Output Supported column of the following tables, note the following:

PCL	<p><i>PCL</i> denotes PCL printing format.</p> <p>Values in parentheses next to PCL, for example <i>PCL (8V)</i>, are symbol sets. If PCL is not listed for an encoding parameter, the encoding is not supported output. Additional hardware support (font ROM) may be required to get correct output.</p>
PS	<p><i>PS</i> denotes a PostScript printer.</p>
CSV	<p>CSV denotes comma-separated value format.</p> <p>CSV (*) means output is supported but Microsoft Excel may not read the output correctly. To read the output in Excel, use a supported encoding other than those indicated with an asterisk (*).</p>
Flat file	<p><i>Flat file</i> denotes that the output is generated by the WRITE command or that the text encoding is readable with the READ command.</p>

SPF, HTML, Enhanced HTML, and PDF can output all the supported languages and encodings. For SPF, HTML, and Enhanced HTML, SQR uses UTF-8 encoding (Unicode), even if ENCODING and ENCODING-REPORT-OUTPUT is set to non-Unicode encoding in the pssqr.ini/unx file. For PDF, character support is dependent on the fonts listed in the [PDF Fonts] section of the pssqr.ini/unx file.

Data processing of Arabic/Hebrew and other languages normally written in right-to-left order is supported, but SQR for PeopleSoft does not generate reports in right-to-left page order.

Arabic

The following table lists the supported encodings for the Arabic character set:

Encoding Parameter	Description	Output Support
CP720	Arabic - Transparent ASMO	PCL (8V), Flat file
CP708	ASMO708	Flat file
CP20240	IBM EBCDIC - Arabic	Flat file
CP28596	ISO 8859-6 (Arabic)	CSV, Flat file
Arabic	ISO 8859-6 (Arabic)	PCL (11N), CSV, Flat file
CP10004	Macintosh Arabic	CSV (*), Flat file
CP1256	MS Windows Arabic	PCL (9V), CSV, Flat file
CP864	MS-DOS Arabic	CSV (*), Flat file

Baltic

The following table lists the supported encodings for the Baltic character set:

Encoding Parameter	Description	Output Support
CP28594	ISO 8859-4 (Baltic)	CSV, Flat file
ISO-8859-4, Latin4	ISO 8859-4 (Baltic)	4N, CSV (*), Flat file
CP1257	MS Windows Baltic	CSV, Flat file
CP775	MS-DOS Baltic	PCL (19L), CSV (*), Flat file

Celtic

The following table lists the supported encodings for the Celtic character set:

Encoding Parameter	Description	Output Support
ISO-8859-14	ISO 8859-14 (Latin 8)	Flat file

Chinese (Simplified)

The following table lists the supported encodings for the Simplified Chinese character set:

Encoding Parameter	Description	Output Support
CCSID935	IBM EBCDIC 935	Flat file
GB18030	GB 18030-2000	Flat file
GB2312	GB 2312-80	PCL (18C), PS, , CSV, Flat file
HZ	HZ GB2312-80	Flat file
CP936	MS Windows Schinese/MS-DOS Schinese (GBK)	PCL (18C), PS, CSV, Flat file
EUC-CN	Simplified Chinese EUC	PS, CSV (*), Flat file

Chinese (Traditional)

The following table lists the supported encodings for the Traditional Chinese character set:

Encoding Parameter	Description	Output Support
Big5	Big5	PCL (18T), PS, CSV, Flat file
CCSID937	IBM EBCDIC 937	Flat file
CNS-11643-1986	CNS-11643-1986	CSV (*), Flat file
CNS-11643-1992	CNS-11643-1992	CSV (*), Flat file
GB12345	GB12345	Flat file

Encoding Parameter	Description	Output Support
CP10002	Macintosh Traditional Chinese	Flat file
CP950	MS Windows Tchinese/MS-DOS Tchinese (Big5)	PCL (18T), PS, CSV, Flat file
EUC-TW	Traditional Chinese EUC	PS, CSV (*), Flat file

Cyrillic

The following table lists the supported encodings for the Cyrillic character set:

Encoding Parameter	Description	Output Support
CP20880	IBM EBCDIC - Cyrillic (Russian)	Flat file
CP21025	IBM EBCDIC - Cyrillic (Serbian, Bulgarian)	Flat file
CP28595	ISO 8859-5 (Cyrillic)	CSV, Flat file
ISOLatinCyrillic	ISO 8859-5 (Cyrillic)	PCL (5T), CSV, Flat file
CP10007	Macintosh Cyrillic	CSV (*), Flat file
CP1251	MS Windows Cyrillic (Slavic)	PCL (5T), CSV, Flat file
CP855	MS-DOS Cyrillic	CSV (*), Flat file
CP866	MS-DOS Russian	CSV (*), Flat file
CP20866	Russian — K018	CSV (*), Flat file
CP21866	Ukranian — K018 — RU	Flat file

Greek

The following table lists the supported encodings for the Greek character set:

Encoding Parameter	Description	Output Support
CP20423	IBM EBCDIC - Greek	Flat file
CP28597	ISO 8859-7 (Greek)	CSV, Flat file
Greek	ISO 8859-7 (Greek)	CSV, Flat file
CP10006	Macintosh Greek 1	CSV (*), Flat file
CP1253	MS Windows Greek	CSV, Flat file
CP737	MS-DOS Greek 437G	CSV (*), Flat file
CP869	MS-DOS Modern Greek	CSV (*), Flat file

Gurmukhi

The following table lists the supported encodings for the Gurmukhi character set:

Encoding Parameter	Description	Output Support
CP10010	Macintosh Romanian	CSV (*), Flat file

Hebrew

The following table lists the supported encodings for the Hebrew character set:

Encoding Parameter	Description	Output Support
CP38598	ISO 8859-8 (Hebrer Logical Ordering)	CSV, Flat file
Hebrew	ISO 8859-8 (Hebrer Logical Ordering)	CSV, Flat file
CP28598	ISO 8859-8 (Hebrer Visual Ordering)	Flat file
CP10005	Macintosh Hebrew	CSV (*), Flat file

Encoding Parameter	Description	Output Support
CP1255	MS Windows Hebrew	CSV, Flat file
CP862	MS-DOS Hebrew	CSV (*), Flat file

Icelandic

The following table lists the supported encodings for the Icelandic character set:

Encoding Parameter	Description	Output Support
CP10079	Macintosh Icelandic	CSV (*), Flat file
CP861	MS-DOS Icelandic	CSV (*), Flat file

Japanese

The following table lists the supported encodings for the Japanese character set:

Encoding Parameter	Description	Output Support
CP21027	Ext Alpha Lowercase	Flat file
CCSID-1027	IBM EBCDIC - Japanese	CSV (*), Flat file
CP20290	IBM EBCDIC - Japanese Kana Extension	Flat file
CCSID939, EBCDIK1027	IBM EBCDIC MBCS-HOST - Japanese (1027+ 0300)	CSV (*), Flat file
CCSID-290	IBM EBCDIK - Japanese Kana Extension	CSV (*), Flat file
CCSID930, EBCDIK290	IBM EBCDIK MBCS-HOST - Japanese (290+0300)	PDF, CSV (*), Flat file
CCSID-942	IBM MBCS-PC OS2 (1041+941) - Japanese	CSV (*), Flat file
ISO-2022-JP	ISO 2022-JP	CSV (*), Flat file

Encoding Parameter	Description	Output Support
EUC-J, JEUC	Japanese EUC	PS, CSV (*), Flat file
JIS_X_0201	JIS X 0201	CSV (*), Flat file
JIS_X_0208	JIS X 0208	CSV (*), Flat file
CP10001	Macintosh Japanese	Flat file
Shift-JIS, SJIS	MS Windows Japanese/MS-DOS Japanese	PCL (19K), PS, CSV, Flat file
CP932	MS Windows Japanese/MS-DOS Japanese	PCL (19K), PS, CSV, Flat file

Korean

The following table lists the supported encodings for the Korean character set:

Encoding Parameter	Description	Output Support
ISO-2022-KR	ISO-2022-KR	CSV (*)
EUC-KR	Korean EUC	PS, CSV (*), Flat file
CP1361	Korean Johab	PS, CSV (*), Flat file
Johab	Korean Johab	PS, CSV (*), Flat file
CP10003	Macintosh Korean	Flat file
CP949	MS Windows Korean/MS-DOS Korean	PS, CSV, Flat file

Latin

The following table lists the supported encodings for the Latin character set:

Encoding Parameter	Description	Output Support
CP20277	IBM EBCDIC - Denmark/Norway	Flat file
CP20278	IBM EBCDIC - Finland/Sweden	Flat file
CP20297	IBM EBCDIC - France	Flat file
CP20273	IBM EBCDIC - Germany	Flat file
CP20871	IBM EBCDIC - Icelandic	Flat file
CP500	IBM EBCDIC - International	CSV (*), Flat file
CP20280	IBM EBCDIC - Italy	Flat file
CP20833	IBM EBCDIC - Korean Extended	Flat file
CP20284	IBM EBCDIC - Latin America/Spain	Flat file
CCSID1047	IBM EBCDIC - Latin1/Open System	Flat file
CP875	IBM EBCDIC - Modern Greek	CSV (*), Flat file
CP870	IBM EBCDIC - Multilingual /ROECE (Latin2)	CSV (*), Flat file
CP20285	IBM EBCDIC - United Kingdom	Flat file
CP20269	ISO 6937 Non-Spacing Accent	Flat file
CP28591	ISO 8859-1 (Latin 1)	CSV (*), Flat file
ISO-8859-1, Latin1	ISO 8859-1 (Latin 1)	PCL (0N), PS, CSV, Flat file
ISO-8859-15	ISO 8859-15 (Latin 9)	CSV (*), Flat file
ISO-8859-2, Latin2	ISO 8859-2 (Central Europe)	PCL (2N), CSV, Flat file

Encoding Parameter	Description	Output Support
CP28593	ISO 8859-3 (Latin 3)	CSV (*), Flat file
CP10082	Macintosh Croatia	CSV (*), Flat file
CP10029	Macintosh Latin2	CSV (*), Flat file
CP1252	MS Windows Latin1	PCL (19U), PS, CSV, Flat file
CP850	MS-DOS Multilingual Latin1	PCL (12U), CSV (*), Flat file
CP860	MS-DOS Portugese	Flat file
CP20261	T.61	Flat file

Latin (Canadian French)

The following table lists the supported encodings for the Latin (Canadian French) character set:

Encoding Parameter	Description	Output Support
CP863	MS-DOS Canadian French	CSV (*), Flat file

Latin (Central European)

The following table lists the supported encodings for the Latin (Central European) character set:

Encoding Parameter	Description	Output Support
CP28592	ISO 8859-2 (Central Europe)	CSV, Flat file
CP1250	MS Windows Central European	PCL (9E), CSV, Flat file

Latin (Southeast European)

The following table lists the supported encodings for the Latin (Southeast European) character set:

Encoding Parameter	Description	Output Support
ISO-8859-3, Latin3	ISO 8859-3 (Latin 3)	CSV, Flat file

Latin (US English)

The following table lists the supported encodings for the Latin (US English) character set:

Encoding Parameter	Description	Output Support
CP20105	IA5 IRV International Alphabet No.5	Flat file
EBCDIC, CP037	IBM EBCDIC - US/Canada	PS, CSV (*), Flat file
CP037	IBM EBCDIC - US/Canada	CSV (*), Flat file
CP437	MS-DOS US	CSV (*), Flat file
ASCII, ANSI	US-ASCII	PCL (0U), PS, CSV, Flat file

Latin (Western European)

The following table lists the supported encodings for the Latin (Western European) character set:

Encoding Parameter	Description	Output Support
CP10000	Macintosh Roman	CSV (*), Flat file

Malayalam

The following table lists the supported encodings for the Malayalam character set:

Encoding Parameter	Description	Output Support
CP10017	Macintosh Malayalam	Flat file

Nordic

The following table lists the supported encodings for the Nordic character set:

Encoding Parameter	Description	Output Support
ISO-8859-10, Latin6	ISO 8859-10 (Latin 6)	PCL (6N), Flat file
CP865	MS-DOS Nordic	CSV (*), Flat file

Slavic

The following table lists the supported encodings for the Slavic character set:

Encoding Parameter	Description	Output Support
CP852	MS-DOS Slavic	PCL (17U), CSV (*), Flat file

Symbol

The following table lists the supported encodings for the Symbol character set:

Encoding Parameter	Description	Output Support
Adobe-Symbol-Encoding	Adobe Symbol Encoding	Flat file
CP10008	Macintosh RSymbol	Flat file

Thai

The following table lists the supported encodings for the Thai character set:

Encoding Parameter	Description	Output Support
CP20838	IBM EBCDIC - Thai	Flat file
Thai	ISO 8859-11 (Thai)	CSV, Flat file
CP874	MS Windows Thai/MS-DOS Thai	CSV, Flat file

Turkish

The following table lists the supported encodings for the Turkish character set:

Encoding Parameter	Description	Output Support
CP20905	IBM EBCDIC - Turkish	CSV (*), Flat file
CP1026	IBM EBCDIC - Turkish (Latin 5)	CSV (*), Flat file
CP28599	ISO 8859-9 (Latin 5)	CSV, Flat file
ISO-8859-9, Latin5	ISO 8859-9 (Latin 5)	PCL (5N), CSV, Flat file

Encoding Parameter	Description	Output Support
CP10081	Macintosh Turkish	CSV (*), Flat file
CP1254	MS Windows Turkish	PCL (5T), CSV, Flat file
CP857	MS-DOS Turkish	PCL (9T), CSV (*), Flat file

Unicode

The following table lists the supported encodings for the Unicode character set:

Encoding Parameter	Description	Output Support
Java	Java Encoding (represents Unicode in US-ASCII)	Flat file
Big-Endian	Unicode Big-Endian Order	CSV (*), Flat file
BMP	Unicode BMP	Flat file
Little-Endian	Unicode Little-Endian Order	CSV, Flat file
UCS-2, UCS2	Unicode UCS-2 Encoding	CSV (*), Flat file
UTF-8, UTF8	Unicode UTF-8 Encoding	CSV (*), Flat file
UTF8-EBCDIC	UTF-8 EBCDIC	Flat file

Vietnamese

The following table lists the supported encodings for the Vietnamese character set:

Encoding Parameter	Description	Output Support
CP1258	MS Windows VietNam	CSV, Flat file

Understanding PS/nVision Reporting in a Global Implementation

In PS/nVision, you can specify the language of the user who is designing or requesting a report, the language of the user who makes a report request, and the language of the ultimate audience of a report (the person or group of people for whom the report was generated). The primary goal of these features is to allow a single layout to produce several instances of a report in a requested alternate language or group of languages. To that end, the features are designed to present reports to end users in their preferred languages. Thus, there is a distinction between the designer's language and the user's language.

Most of the strings in PS/nVision output are fetched from the Strings table and they are delivered with the report instance. However, many labels used in macros and dialog boxes have to be translated directly in the file, as they cannot be fetched at runtime. When a PS/nVision report happens to contain labels in macros or dialog boxes, the layout needs to be translated in Excel, and a separate copy of the layout maintained for each language. In this case, each language has its own version of the file, and the same rules for locating the appropriate language of the PS/nVision layout spreadsheet files are used as for Crystal Reports described above. Each PS/nVision layout and drill-down directory can contain a subdirectory for each language. PS/nVision searches the appropriate directory for the user's current language, and performs the appropriate fallback if a translation is not found.

Language-sensitive features of PS/nVision can be broken into two distinct parts:

- Design-Time Features** These are features related to designing PS/nVision layouts.
- Runtime Features** These are features activated when a user makes a PS/nVision report request.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PS/nVision

PS/nVision Design-Time Globalization Features

The following design-time features of PS/nVision adapt to the current user's language and enable you to build a layout that can produce reports in multiple languages

- Column Headings** When the user selects a column from a list, the list appears in the user's language (unless database field names are requested). Wherever the heading is stored in the layout, it's stored in the base language.
- Tree Names** When prompting for tree names or presenting a tree description—for example, when adding tree criteria for a field—the tree description appears in the user's language. Depending on the structure of your tree, this may require adding related language records to one or more prompting views.
- String Variables** Layouts typically contain a considerable amount of constant text, such as the column headings Last Year to Date and Current Budget. PS/nVision provides the option to build multilingual layouts where these text strings are replaced by specially formatted strings whose user-language equivalent can be retrieved from the PeopleTools Strings table. This enables a single PS/nVision layout definition to be used in multiple languages without duplicating the layout itself. These string names resemble user-defined PS/nVision variables.

PS/nVision translates only the strings that occupy a layout cell and are in the following format:

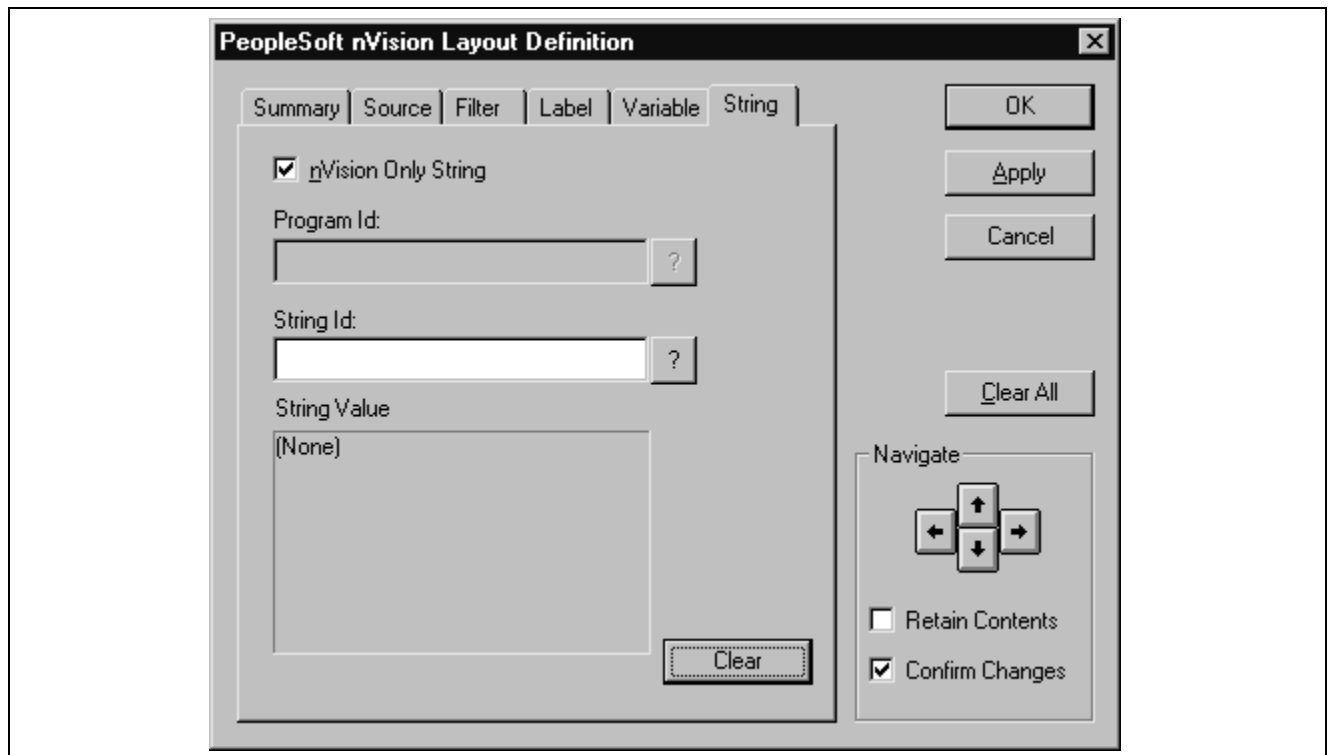
`%.name,program%`

Where *name* is the string name and *program* is the optional group that is used to collect a common set of strings used on similar reports. You can look up the *name* in the Strings table using the Strings Table utility.

Note. This is similar to the approach used by PeopleSoft application SQRs. Names are case sensitive.

Setting Up String Criteria

Use the PeopleSoft nVision Layout Definition dialog box: String tab to set up string criteria for the current cell selection.



PeopleSoft nVision Layout Definition dialog box: String tab

To set up string criteria for the current cell selection:

1. Open the PeopleSoft nVision Layout Definition dialog box.
2. Select the String tab.

The String tab in the PeopleSoft nVision Layout Definition dialog box appears.

Use this tab to insert a string from the Strings table into a cell in your layout.

3. Clear the nVision Only String box, if appropriate.

By default, this tab displays only the strings that were created for use with PS/nVision—those with a program ID of NVISION. If you want to select from all available strings, clear the nVision Only String box.

4. If the nVision Only String box is cleared, select a Program ID.

Note. If the nVision Only String check box is selected, the program ID is *NVISION*.

5. In the String ID field, select the ID of the string you want to insert.
Select any of the strings assigned to the program ID that you specified.
6. Click Apply to save your changes and define string criteria for a different cell, or click OK to save your changes and close the dialog box.
If you clicked Apply, and you want to reuse all or part of the criteria you just applied, select the Retain Contents box. This preserves all the dialog box information when you navigate to a new cell selection. Then repeat steps 3 through 5 to define additional string criteria.

See Also

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” Understanding the Strings Table and Language-Sensitive Text in Reports, page 104](#)

PS/nVision Run-Time Language Features

The following features are activated when a user makes a PS/nVision report request:

Path Search

As PS/nVision looks for a file (such as an Excel spreadsheet, layout, or template), it steps through the directories listed in the appropriate paths. The paths are defined in different ways for the Windows client and the Web. In the Windows client, the paths are defined in PeopleSoft Configuration Manager. In the Web, the paths are defined in the [nVision] section of the PeopleSoft Process Scheduler configuration file (psprcs.cfg).

Before looking in each directory, PS/nVision looks in a subdirectory named by the user’s language code. If the file is not there, or if the appropriate directory does not exist, PS/nVision looks in the directory named in the path. If the file is not there, PS/nVision moves to the next directory in the path and repeats the process.

For example, when searching for NVSUSER files for a user with France as the preferred language and the defined path for macros is %PS_HOME%\EXCEL, PS/nVision searches for %PS_HOME%\EXCEL\FRA directory first. If the files are not found, PS/nVision uses the files in the %PS_HOME%\EXCEL.

This feature supports users who need or prefer different layouts and for users of different languages. For example, it may be necessary to create an earnings report differently in Germany and France because differences in accounting rules or management requirements dictate different criteria and formatting. So, if a French user runs a PS/nVision report, PeopleTools first looks in the FRA directory under the PS/nVision directory that is defined for an installation. If the required report layout isn’t found in the FRA directory, PeopleTools uses the generic report layout in the base PS/nVision directory.

Request Language

The PS/nVision Report Request page includes a Language Template text box on the Advanced Options page. In the Foreign Language Translation field, enter an alternate language code to automatically generate a translated report. If you are applying a scope to the report request, enter a string containing one or more PS/nVision variables.

If you enter one or more variables in the Foreign Language Translation field, then the value of each variable is interpreted at runtime to get the language code for each report instance. This enables retrieval of the language code from a tree node or value table that is associated with the values in the scope.

The syntax of this variable is as follows:

```
%DES.[scopefield].{detailfield|.nodefield|detailfield.nodefield}%
```

Note that the field names required vary, depending on the type of scope you're using. However, the periods between the values must always exist—except in the case of a trailing period. The *scopefield* parameter is optional. If you don't specify one, PS/nVision uses the first scope field you defined. If you want to use a field other than the first one defined in the scope, then you must specify which one.

You can provide a *detailfield*, *nodefield* or both. PS/nVision uses the appropriate field type based on your scope. Consequently, specifying one of each type enables you to change your scope definition without necessarily changing this variable. The detail and node table fields you specify should store PeopleSoft language codes and can be found on the same node or detail value table associated with the *scopefield*. Typically, the field name used is *LANGUAGE_CD*.

The follow example shows a common implementation of this variable (note the inclusion of the extra period):

```
%DES..LANGUAGE_CD.LANGUAGE_CD%
```

If your scope consists of multiple scope fields (PRODUCT and BUSINESS_UNIT), your variable might look like this:

```
%DES.BUSINESS_UNIT.LANGUAGE_CD.LANGUAGE_CD%
```

If you don't include a *scopefield* or *detailfield* value, it's important that you still include the extra periods that follow those values. For example:

```
%DES...nodefield%
%DES.scopefield..nodefield%
```

However, if you don't include a *nodefield* there's no need to include the trailing period after the *detailfield*. For example:

```
%DES..detailfield%
```

Using a *scopefield* to drive the language of the PS/nVision report enables you to run a single report in multiple languages. Each time the contents of the *scopefield* changes, PS/nVision resets the language of the report and reloads the strings and other language-sensitive objects in the new language. This enables you to create a PS/nVision report in multiple languages so that the report can be separated and delivered to multiple recipients, each of whom may have a different language preference.

Labels

When retrieving node or detail row and column labels, PS/nVision uses the appropriate alternate language record, if one has been defined through PeopleSoft Application Designer, to get labels in the user's language.

PS/nVision Variables

When retrieving a descriptive value for a PS/nVision variable, such as *Business Unit Description*, PS/nVision determines whether the table being queried has a related language record and if the field being retrieved is on that record. If so, and if the user does not use the base language, PS/nVision retrieves the value from the related language record or, if the row doesn't exist on the related language record, from the base record.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PS/nVision, “Running PS/nVision Reports in the Microsoft Windows Client,” Creating Requests

Enterprise PeopleTools 8.45 PeopleBook: PS/nVision, “Defining Report Scopes”

International Versions of Microsoft Excel

Microsoft Excel is available in numerous languages. PeopleTools is designed to work with all international editions of Excel as long as the base version of Excel is supported by that version of PeopleTools. For example, if your current PeopleTools version supports Excel 2000, then all language editions of Excel 2000, such as Japanese Excel 2000, are supported by the PeopleSoft system.

Note. Downloaded data such as numbers and dates are formatted in the Excel spreadsheet according to the regional settings on the user's machine.

Note. If you plan to download grids from PeopleSoft pages to Microsoft Excel 97 and you want to use the character set defined in the user language—that is, you want to use a character set other than the default UTF-8 character set—then you must select the *EXCEL97* option on the Personalizations page and set its option value to *Y*. This option is only recommended for non-English users who use Excel 97. It is not recommended for Excel 2000 or later versions.

See Also

PeopleTools 8.45 Hardware and Software Requirements Guide

CHAPTER 9

Developing Global Applications

When you implement a global PeopleSoft application, a number of factors related to globalization can affect system performance and the efficiency of the implementation effort.

This chapter discusses how to:

- Develop easily translatable applications.
- Design global-ready pages.

Developing Easily Translatable Applications

When designing application objects, especially pages, follow these guidelines to ensure easy translation:

- Avoid using page-based text in page element labels and in other labels that can be derived from field labels (such as query heading names and string definitions in the Strings table [STRINGS_TBL]).

Overriding field labels on the page with page-based text associates a new label with a field that is specific to the current page. The new label cannot be shared with other pages in the system, so if the new label is used multiple times, it requires translation for each occurrence. Instead, associate the new label with the field that it is labeling, and choose your new label for the occurrence of the field on your page. That way, the label can be used on other pages without having to be translated again.

- Design pages with enough room to accommodate object labels that become longer when translated.

A good guideline if developing in English is to allow at least 15 percent extra space between a label and its field, and 50 percent extra space for abbreviations.

- Complete page layout early in the development cycle.
- Because the positions of fields and other objects on a page are stored individually for each language, you might need to reapply page layout changes after the page has been translated.

Translate field labels first. Translated field labels appear throughout the system in page element labels, search dialog box field labels, record field names, Strings table definitions, query heading labels, and so on. Translating these labels before translating other objects helps you to build a glossary of terms for your application and give you a head start in translating each page, as the majority of text on a page is derived from field labels.

Selective Translation

If you add custom functionality or data tables to a PeopleSoft application, you might want to translate the new PeopleTools objects and data fields for display in multiple languages. PeopleTools is designed to permit selective translation of the elements in an application database.

Because the system displays base-language versions of any language-sensitive elements for which no translation is provided, you can translate only the pages, menus, messages, and so on that are required by users who are working in non-base languages. Often, you can easily determine if the page will be accessed only by administrative staff who share a common language with developers, or if it will be deployed widely and will therefore require translation.

Just because an object is not translated into a user's preferred language doesn't mean that the user can't access that object—the user sees the object in the base language of the system.

Designing Global-Ready Pages

You can simplify a global development project and make a global system easier to maintain by making base-language pages as translation-ready as possible. Reduce or eliminate textual elements that are maintained on the page definition and instead derive text strings from other PeopleTools objects, such as field labels or messages. This way, a translator needs to translate a text string only once, and the new translation takes effect across all pages where that string is referenced.

To design global-ready base language pages:

- Use page control labels that derive from field descriptions.

On the Label tab of the Page Field Properties dialog box in PeopleSoft Application Designer, use the RFT Short or RFT Long setting whenever possible. This causes the control labels, as well as the button tooltip text, to be derived from the language-sensitive descriptions that are stored in the field definition. Avoid using any other non-language-sensitive text, such as page-based floating text labels, on the page.

- Associate group boxes with record fields.

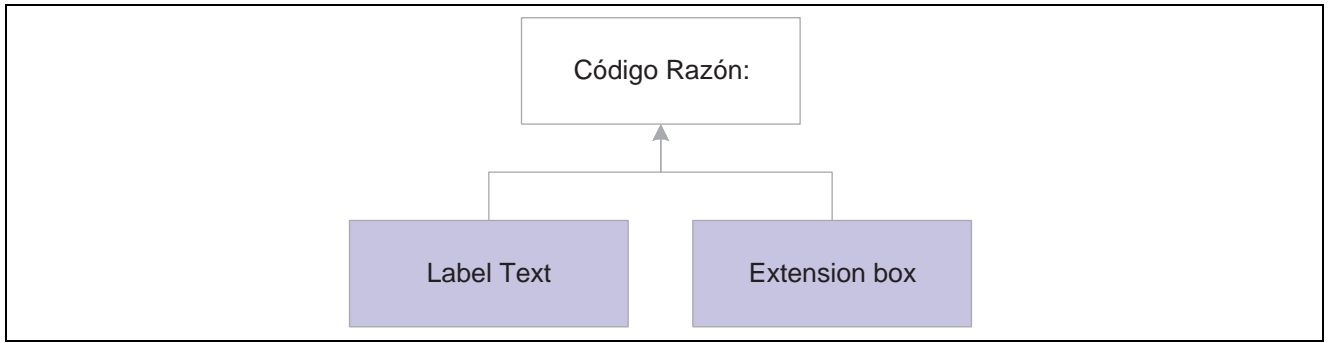
Use the Page Field Properties dialog box in PeopleSoft Application Designer to associate group boxes with record fields. In this way, the label for the group box can be derived from field's RFT Short or RFT Long value. The only effect of this association is the label derivation. It has no other effect on the page's operation.

- As much as possible, complete and freeze the layout of the pages early in your development cycle.

Adding and removing page objects and PeopleCode on a page takes effect across all languages; however, the layout of a page is language-specific. After you start translating pages, any alignment or page layout changes you make to the base language page must be reapplied to each translation of the page.

- Size and arrange page controls so that there is enough space to accommodate data in non-base languages.

English strings for both labels and data tend to be shorter than strings in other languages. As you work with pages, you will notice that when you select a field, the dotted boundary box has two components—the surround box for the label, and an extension box to the right of the label. This extension box size is typically 10 percent of the label size and is a useful guide as to the minimum amount of space that you should leave between a field and its label to allow for expansion during translation. Ensure that there is enough space between the field and the label, so that the extension box doesn't overlap the field or other page controls. The following graphic shows the label text and extension box:



Page control label and extension box

PART 3

Working With PeopleSoft Applications in Multiple Languages

Chapter 10
Using Related Language Tables

Chapter 11
Working With Language-Sensitive Application Data

CHAPTER 10

Using Related Language Tables

This chapter provides an overview of related language tables and discusses how to:

- Install PeopleSoft-provided translations.
- Swap the base language.
- Create related language tables.
- Create related language views.

Understanding Related Language Tables

This section discusses:

- Related language tables overview.
- Related language table structure.
- How related language tables store transactions.
- How related language tables are used.

Related Language Tables Overview

PeopleTools can store multiple translations of application data and PeopleTools objects in a single database. Each PeopleSoft database has a single *base language*. The base language of a new PeopleSoft database upon installation is always English, but can be changed by an administrator. The base language should be the language most commonly used by application users, and is the language in which data is stored in the core PeopleSoft tables known as *base language tables*.

All PeopleTools objects, such as pages, fields and queries, can be maintained in multiple languages. Descriptions of application data elements, such as departments, locations and job codes, can also be maintained in multiple languages.

The key to maintaining this data in multiple languages is the use of special tables known as related language tables. A *related language table* stores descriptions and other language-sensitive elements in all languages other than the base language of the database. In this way, while any table in the database can store data in the base language of that database, only tables that have related language tables can maintain the same data in multiple languages simultaneously. For example, it is unlikely that you will want to maintain the descriptions of your general ledger journal lines in multiple languages – the sheer volume of the journal lines in most systems would preclude any effort to maintain translations of their descriptions. The cost of hiring a translator to translate each journal line would be prohibitive, and in most cases only the person entering the journal line, and possibly their supervisor, would be likely to want to view it again. However for frequently used values, such as a chart of accounts, many users across your entire organization would often need to refer to this data, and so you would most likely want to maintain the descriptions of each ChartField entry in each language spoken by your users.

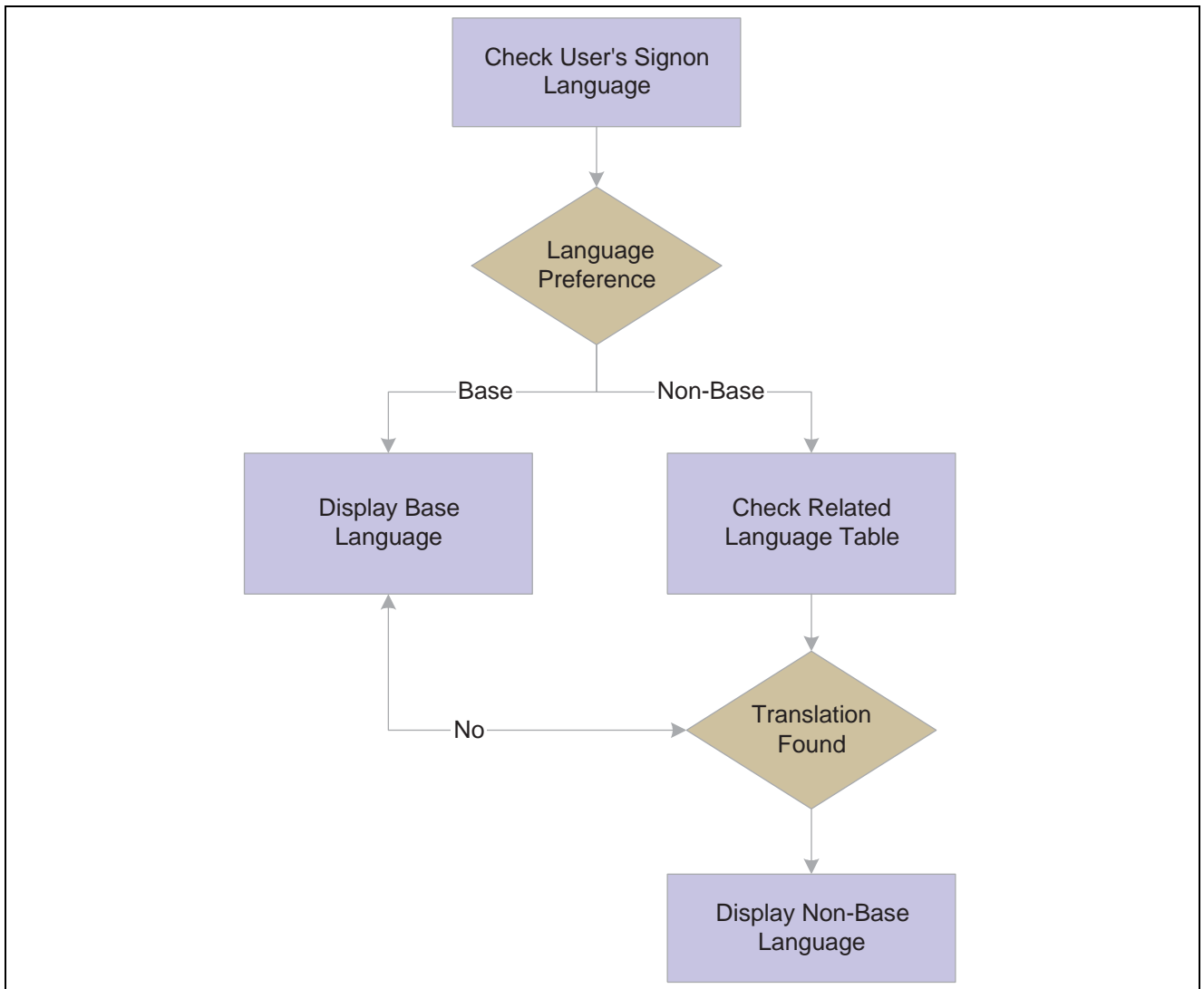
In this case, you would not need a related language table for your Journal Lines table, as you would be maintaining Journal Line descriptions in a single language only, and this would be in the base table, but you would need a related language table for each of your ChartField tables.

When the system displays a language-sensitive field value, it retrieves the text from either the base table or the related language table, depending on the following:

- The current language preference.
- Whether any translated rows for the field are found in the related language table.

The *language preference* refers either to the PeopleSoft Pure Internet Architecture sign-in language or, in the case of PeopleSoft Application Designer, to the language preference as determined by the PeopleSoft Configuration Manager language setting.

If the current language preference is the system's base language, the text is retrieved from the base table. If the language preference is a non-base language, then the system looks for a translation of the text in the related language table. If it finds a translation, it displays the translated text; if no translation exists, the system uses the text in the base table. This enables developers to selectively translate portions of the system, while keeping the system fully functional at all times, even if not all rows have been translated. The following flowchart shows this flow of execution:



System display of base and non-base language text

Synchronization of related language tables with their base tables is automatically maintained by PeopleTools. When you translate a language-sensitive field, the system adds a new row to the related language table. When you delete a row from a base table, any child rows in the related language table are deleted. The primary responsibility of the application developer in the language architecture is to define and maintain the related language tables.

Note. PeopleTools automatically handles the storage and display of data in related language tables and their synchronization with corresponding entries in the base language tables based on each user's language preference. However, PeopleTools cannot automatically translate data! The PeopleSoft system provides translations of PIA components and user interfaces for most application products into several languages. Also, translations of several key application data tables, such as country and currency codes, are provided by the PeopleSoft system. However, you will need to plan a way to translate application data entered by your users that needs to be maintained in multiple languages. Some users who are familiar with several languages may want to translate the data elements they enter themselves using the Multilingual Support feature. In other cases, you may want to bring in translators periodically to translate key elements into each language used in your organization. PeopleTools provides an architecture to store these translations, but does not perform "machine translation" or provide other automated linguistic translation technologies.

See Also

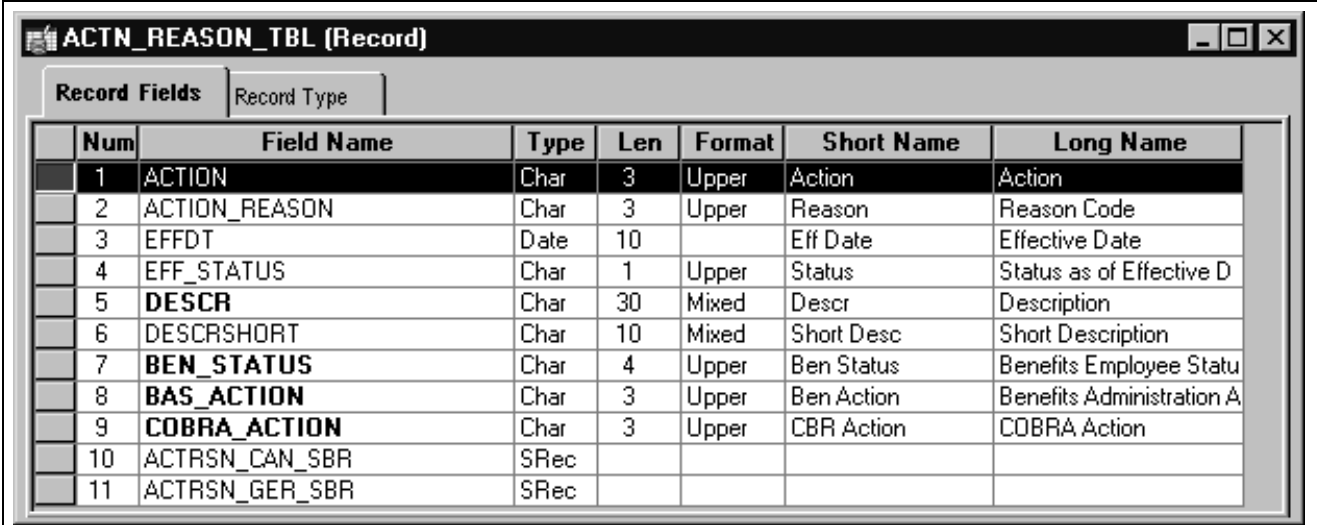
Chapter 2, “Controlling International Preferences,” page 11

Chapter 11, “Working With Language-Sensitive Application Data,” Editing Data in Multiple Languages, page 170

Related Language Table Structure

Related language tables store multiple translations of language-sensitive fields from an associated base table. For example, description (DESCR) and short description (DESCRSHORT) fields are commonly language sensitive.

Each table in your database that is not a related language table is, by definition, a base language table. The Action Reason table, ACTN_REASON_TBL, is one example of a base table. Note that it has DESCR and DESCRSHORT fields which would typically be language-sensitive. All other fields in the table are translate value or prompt table editable codes and are therefore not language sensitive. In the following example, the developer has decided that it is likely that each user would need to see the description of the Action Reason in their own language, and so a related language table is needed to store translations of DESCR and DESCRSHORT.



Num	Field Name	Type	Len	Format	Short Name	Long Name
1	ACTION	Char	3	Upper	Action	Action
2	ACTION_REASON	Char	3	Upper	Reason	Reason Code
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
7	BEN_STATUS	Char	4	Upper	Ben Status	Benefits Employee Statu
8	BAS_ACTION	Char	3	Upper	Ben Action	Benefits Administration A
9	COBRA_ACTION	Char	3	Upper	CBR Action	COBRA Action
10	ACTRSN_CAN_SBR	SRec				
11	ACTRSN_GER_SBR	SRec				

ACT_REASON_TBL record

The developer has therefore created a new record definition, ACTN_RSN_LANG, to act as the related language table for ACTN_REASON_TBL. Related Language records are defined in Application Designer just like any other record definitions. However, as they often contain many of the same fields as their base language table, it is often quicker to create the related language table by *cloning* the base table. To clone a base table, open the base language table and use the Save As option on the File menu and enter a name for the new record. Once cloned, any fields not required on the related language record can be deleted from the new record.

A related language table must:

- Share all key record fields with its base table.
- Have an additional key record field, which must be LANGUAGE_CD.
- Have language-sensitive fields matching those in the base table. These are typically DESCR and DESCRSHORT, or LONGNAME and SHORTNAME.
- Not contain any fields that are not present on the base table, except the LANGUAGE_CD key field.
- Not contain any non-key, non-language-sensitive fields from the base table.

The related language table, ACTN_RSN_LANG, meets all of these requirements:

Num	Field Name	Type	Len	Format	Short Name	Long Name
1	ACTION	Char	3	Upper	Action	Action
2	ACTION_REASON	Char	3	Upper	Reason	Reason Code
3	LANGUAGE_CD	Char	3	Upper	Lang Cd	Language Code
4	EFFDT	Date	10		Eff Date	Effective Date
5	DESCR	Char	30	Mixed	Descr	Description
6	DESCRSHORT	Char	10	Mixed	Short Desc	Short Description

ACT_RSN_LANG record

The base table must be explicitly associated with its related language table.

To associate a base table with its related language table:

1. Open the Record Properties dialog box for the base table record.
2. Click the Use tab.
3. Under Record Relationships, in the Related Language Record field enter the name of the related language table.

Note. There is a one to one relationship between the base record and the related language record.

4. Click OK.

How Related Language Tables Store Translations

Base tables store language-sensitive descriptions in the database's base language. Related language tables store translations of these descriptions in non-base languages. For each row in the base table, there can be zero rows or one row in the related language table for each non-base language.

For example, the following table shows a row of data from the ACTN_RSN_TBL table.

Action	Action_Reason	Effdt	Descr	DescrLong
PAY	PRO	01/01/98	Promotion	Promotion

The following table shows child rows of data from the ACTN_RSN_LANG table. For each LANGUAGE_CD, there is a row with appropriate translations of the language-sensitive fields in the parent row.

Action	Action_Reason	Effdt	Language_Cd	Descr	DescrLong
PAY	PRO	01/01/98	ESP	Promoción	Promoción

Action	Action_Reason	Effdt	Language_Cd	Descr	DescrLong
PAY	PRO	01/01/98	FRA	Promotion	Promotion
PAY	PRO	01/01/98	GER	Beförderung	Beförderung

How Related Language Tables Are Used

Related language tables are the primary means of storing language translations in the PeopleSoft system. They store translations of descriptive text for data, such as a standard list of expense categories, and system objects, such as descriptions of fields and records.

Related language tables that store descriptive text for data are generally associated with edit tables that store lists of values that are used for validation and prompting. When a page field prompts against a table that has an associated related language table, and, if the user's language preference is a non-base language, the description fields that have been translated into the current language appear in place of the base language descriptive text.

For example, the following page appears when the user's language preference is Spanish (ESP). The Country field (País) prompts against the Country table (COUNTRY_TBL). The translated description (Francia) comes from the Country table's related language table (COUNTRYTBL_LANG).

Page with translated descriptive text

Many PeopleTools object definitions, including fields, records, and components, also have associated related language tables. This makes maintenance of multiple language PeopleTools objects, and therefore the system user interface, relatively straightforward. The system uses a single mechanism for maintaining translated data and for maintaining the system objects themselves. Field translation is particularly powerful because translated field descriptions appear on search pages and on pages where the descriptive text for the field is used as the field label.

In the following example all the fields on the ADDRESS2_SBP subpage are translated into Spanish:

Field	Base Long Name	Tgt Long Name	Base Short Name	Tgt Short Name
Label	Base Label Text	Tgt Label Text		
Page Text	Base Long Name	Tgt Long Name	Base Short Name	Tgt Short Name
Xlat Value Xlat Eff Date	Base Long Name	Tgt Long Name	Base Short Name	Tgt Short Name
ADDRESS2_SBP				
ADDRESS1				
Lbl - ADDRESS1	Address Line 1	Dirección1	Address 1	Dirección1
ADDRESS2				
Lbl - ADDRESS2	Address Line 2	Dirección2	Address 2	Dirección2
ADDRESS3				
Lbl - ADDRESS3	Address Line 3	Dirección3	Address 3	Dirección3
CITY				
Lbl - CITY	City	Cuidad	City	Cuidad
COUNTRY				
Lbl - COUNTRY	País	País	País	País
VIEW_RESUME				
ZIP				
Lbl - ZIP	Postal Code	Cd.Postal	Zip	Cd.Postal

Translation of fields in ADDRESS2_SBP

Because the fields are translated, the translated versions of field labels appear when a user whose language preference is Spanish accesses a page that shows the long name or short name of any of these fields. No additional development work on the part of the developer is required.

Of course, whenever the base labels change, you must also update the translations of the object names.

See Also

[Chapter 11, “Working With Language-Sensitive Application Data,” page 169](#)

[Chapter 15, “Translating Application Definitions,” page 217](#)

Installing PeopleSoft-Provided Translations

This section provides an overview of translation installation, and discusses how to:

- Add translations to an existing database.
- Upgrade your customizations into a multilingual database.

Understanding Translation Installation

In many cases, PeopleSoft provides translations of the user interface and key application data for many of our application products. Except for PeopleTools, these translations are shipped on a separate CD-ROM for each product line known as the "Global Multi-Language CD-ROM". If you did not receive this CD-ROM in your kit, you may not have licensed the appropriate translations from PeopleSoft. Contact Customer Care to obtain the CD. Translations for PeopleTools itself are shipped on the PeopleTools CD — no Global Multi-Language CD-ROM is required for PeopleTools.

Typically you would install these translations when you first install your database. However, if you did not install the PeopleSoft-provided translations during your initial install, this section describes the steps to follow to add translations that you have licensed from PeopleSoft to an already-existing database.

Note. You should install the PeopleSoft Global Multi-Language CD-ROM at the same time that you install the English release.

To add translations licensed from PeopleSoft, you must determine how you will perform your upgrade. Choose the most appropriate option:

- Add the translations provided on the PeopleSoft Global Multi-Language CD-ROM to your existing PeopleSoft database.

This option is available only if you have not made any customizations to existing PeopleTools-based objects in your existing database.

PeopleTools-based objects include any objects that you maintain using PeopleSoft Application Designer or PeopleSoft Tree Manager, such as pages, fields, and menus. If you have added only new PeopleTools objects to your database, without modifying existing definitions, then you can use this option.

Adding or customizing system data, such as account codes, departments, and locations, is not considered a PeopleTools object change. If you have performed only these types of transactions against your existing database, and you have not modified any objects using PeopleSoft Application Designer or PeopleSoft Tree Manager, then you can use this option.

- Upgrade your customizations to a multilingual database.

If you have customized the PeopleTools objects in your database, you cannot simply add the translations contained on the PeopleSoft Global Multi-Language CD-ROM to your existing database. Instead, you must create a new database containing all of your licensed translations and use the upgrade process to copy customizations from your existing database to your newly installed database.

Adding Translations to an Existing Database

This section explains how to load translations by adding them to an existing database. This method of loading translations may not be available to you, depending on the customizations you have performed on your existing database. Be sure to read the overview for this section before loading translations.

To add translations to an existing database:

1. Load your PeopleSoft Global Multi-Language CD-ROM.

See *PeopleSoft Installation and Administration Guide*, "Setting Up the File Server."

2. Rerun the steps for setting up the batch environment.

If you are running a Windows NT or Windows 2000 process scheduler (*PSNT*), then you must rerun the steps for "Setting Up the Batch Environment" documented in your *PeopleSoft Installation and Administration Guide*.

Because the PeopleSoft Global Multi-Language CD-ROM contains translations for your Crystal Reports and PS/nVision layouts, you must transfer those translations to your Windows NT or Windows 2000 process scheduler.

If you run only non-Windows NT or Windows 2000 process schedulers, you do not need to complete this step. This step also does not need to be performed for UNIX batch servers because the only objects that don't contain translations in the database are Crystal and PS/nVision reports, both of which are supported on only Windows NT and Windows 2000 batch servers.

3. Generate database setup from PeopleSoft Data Mover.

Run PeopleSoft Data Mover in bootstrap mode by logging in with your database's access ID. From PeopleSoft Data Mover, select File, Database Setup, and follow the wizard step by step to generate the database setup script. Be sure to select "Add New Language" in the "Database Type" selection box in the second step of the Wizard. The generated database setup script appears in the PeopleSoft Data Mover input view. Be sure to enter the same database parameters that you entered the first time you ran the database setup program to install your English database; do not select any language that already exists in your database.

If you choose a new base language, then the generated script will include the command to swap the base language.

4. Run the PeopleSoft Data Mover script.

Because you have already created your database and set up the appropriate database-specific objects, you must perform only the "Run Data Mover Import Scripts" task.

5. Run the VERSION Application Engine program to update your PeopleTools version numbers.

Run the PeopleTools VERSION Application Engine program against your database. This synchronizes the PeopleTools version numbers, taking the new translations into account.

6. Rerun your database audits.

Your database should now be complete. To ensure that it is complete, rerun the database audits. Refer to the section about creating and checking a database in your installation and administration documentation for more information.

See PeopleSoft Installation and Administration Guide”

Upgrading Your Customizations into a Multilingual Database

This section explains how to load translations by upgrading your customizations into a multilingual database. Be sure to read the overview for this section before loading translations.

To upgrade your customizations into a multilingual database:

1. Follow steps one and two in the previous section entitled Adding Translations to an Existing Database.
2. Create and load your new PeopleSoft Global Multi-Language database, which contains all of the languages you've licensed.

Refer to the section on creating a database in your installation and administration documentation for more information.

Keep in mind that you must load the English objects provided by PeopleSoft even if you don't plan to run your system in English. Follow all of the applicable steps in the documentation about creating a database. Do not follow the Swap Base Language step. You should not swap the base language of your new database until you have copied your customizations into it.

3. Copy your customizations into your new PeopleSoft Global Multi-Language database.

Using the Upgrade Copy utility in PeopleSoft Application Designer, copy the objects you have modified from your existing database into your new PeopleSoft Global Multi-Language database.

If you have maintained all of your customizations in PeopleSoft Application Designer projects, you can easily use these projects to copy only the objects you have modified over to your new PeopleSoft Global Multi-Language database.

If you have not maintained your customizations in a project, you may want to use the PeopleSoft Application Designer Upgrade Compare process to compare your customized database against a newly installed, English-only database in order to identify your changes. Remember that for this compare process, your source database is your existing database, and your target database is a newly installed, English-only database.

Once you have identified the objects you have changed, using the Compare process, use Upgrade Copy to copy the objects into your new Multi-Language database. Remember that during the Upgrade Copy process, your source database is your existing English-only database and your target database is the new Multi-Language database.

4. Run the VERSION Application Engine program to update your PeopleTools version numbers.

Run the PeopleTools VERSION Application Engine program against your database. This will synchronize the PeopleTools version numbers, taking the new translations into account.

5. Swap your base language.

If you have not already swapped your base language and you decide to run your database in a non-English base language, swap the base language of your database.

Your database should now be complete. To ensure that it is complete, rerun the database audits, as documented under the heading "Check Database" in chapter 5, Creating a Database, of the PeopleSoft Installation and Administration Guide.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Application Designer, "Upgrading with PeopleSoft Application Designer"

[Chapter 10, "Using Related Language Tables," Swapping the Base Language, page 158](#)

PeopleSoft Installation and Administration Guide

Swapping the Base Language

This section provides an overview of base language swapping, lists prerequisites, and discusses how to:

- Run the swap audit report.
- Check space and resources.
- Run the SWAP_BASE_LANGUAGE Data Mover command.

Understanding Base Language Swapping

As delivered, the PeopleSoft database has a base language of English. The system has English data in its base language tables and, if you have licensed and installed the PeopleSoft-provided translations, non-English data in the related language tables. To choose a language other than English as your base language, you must move all English data to the related language tables (with an appropriate language code) and move all of the data in your new base language to the base tables. This process is called *swapping the base language* of your database. If you later decide to change base languages again, you must go through the same process.

PeopleSoft Data Mover provides the following command to automate this process:

```
SWAP_BASE_LANGUAGE <target language>;
```

To swap the base language of your database, you must identify the language to swap in as the new base language. PeopleSoft Data Mover finds all the affected tables and loads the appropriate language data into them.

This process handles all related language tables in the system: PeopleTools tables and application tables. Changes are committed after each table is swapped; a log file records the process of the swap and can help you troubleshoot any errors during the process.

You can check the existing base language code by looking in the Options page. To navigate to the Options page, select PeopleTools, Utilities, Use, Options. You can't change the Base Language Code setting in this page; it is for display purposes only. To change your base language, use the SWAP_BASE_LANGUAGE PeopleSoft Data Mover command.

Prerequisites

Before swapping the base language of your database, you must have:

- A clean audit report to ensure that there are no data integrity problems.
- Sufficient space and resources to run the swapping process.

Running the Swap Audit Report

The SQR process SWPAUDIT.SQR ensures that there are no data integrity problems that will cause errors during the swap process.

This SQR does not have a run control, and therefore cannot be run with PeopleSoft Process Scheduler. For information on how to run SQRs manually in your environment, refer to the section on creating a database in your PeopleTools installation documentation.

SWPAUDIT.SQR may report one or more of the following errors. Refer to the instructions below to correct these errors before continuing with your swap process, and rerun the audit until it reports no errors.

Warning! Errors reported during the audit process indicate that the swap base language process will fail unless the errors are corrected.

(SWAP-1) Related Language Records That are not Valid Records

The records listed are defined as related language records for one or more base records, but they do not exist in your database.

To fix this problem, do one of the following:

- Create the related language record, as appropriate, based on the keys and translatable fields of the base record.
- Open the base language record in PeopleSoft Application Designer, and remove its association with the related language record in the Object Properties dialog box.

(SWAP-2) The Field LANGUAGE_CD is not a Key in the Following Related Language Record(s)

A record is defined as a related language record, but it does not have LANGUAGE_CD as a component of its key. LANGUAGE_CD must be a key field on every related language record.

To fix this problem, make LANGUAGE_CD into a key field on the related language record.

(SWAP-3) The Following Related Language View(s) Have the Wrong Key Structure Defined

The keys of a view used as a related language record by one or more records do not conform to the requirements for PeopleSoft related language records. The related language record must share all of the same keys as its base language record, plus an additional key (LANGUAGE_CD).

To fix this problem, correct the key structure to ensure that it conforms to this requirement. Then recreate the view in the database.

(SWAP-4) The Following Related Language Table(s) Have the Wrong Key Structure Defined

The keys of a table used as a related language record by one or more records do not conform to the requirements for PeopleSoft related language records. The related language record must share all of the same keys as its base language record, plus an additional key (LANGUAGE_CD).

To fix this problem, correct the key structure to ensure that it conforms to this requirement. Then alter the table in the database to match the new key structure.

(SWAP-5) The Following Related Language Table(s) Have the Wrong Structure Defined

The fields of a table used as a related language record by one or more records do not conform to the requirements for PeopleSoft related language records. The related language record must share all of the same fields as its base language record, plus an additional key (LANGUAGE_CD).

To fix this problem, correct the field structure to ensure that it conforms to this requirement. Then alter the table in the database to match the new field structure.

(SWAP-6) The Following Related Language Table(s) Have the Orphan Row Defined

For each row on the related language record there must be a single row on the base table with matching keys. An orphan row is a row of data on the related language record that does not have a corresponding parent row on the base table. You must delete orphan rows.

To fix this problem, use your platform's query tools to select against the related language table, using the fields listed in the report, where the values do not match a row on the base table. For every row on the report there is an orphan row on the table. Do a SELECT first to ensure that you are getting the same row count as the report then DELETE the selected rows. Here is sample SQL for a Microsoft SQL database.

```
SELECT * FROM PSCONTDEFNLANG A WHERE NOT EXISTS
(SELECT 'X' FROM PSCONTDEFN B WHERE A.ALTCONTNUM = B.ALTCONTNUM
AND A.CONTNAME = B.CONTNAME AND A.CONTTYPE = B.CONTTYPE)

DELETE FROM PSCONTDEFNLANG WHERE NOT EXISTS
(SELECT 'X' FROM PSCONTDEFN B WHERE PSCONTDEFNLANG.ALTCONTNUM = B.ALTCONTNUM
AND PSCONTDEFNLANG.CONTNAME = B.CONTNAME AND PSCONTDEFNLANG.CONTTYPE = B.CONTTYPE)

SELECT * FROM <Related Language Record> A WHERE NOT EXISTS
(SELECT 'X' FROM <Base Record> B WHERE A.<Field Name> = B.<Field Name> ...for each=>
field name listed)
```

(SWAP-7) The Following Base Records do not Have an Unique Index

A base record that has a corresponding related language record must have a unique index; that is, a key field.

To fix this problem, use PeopleSoft Application Designer to open the base record listed in the report and select a field that you want to make a key field. Right mouse click on the field and select Record Field Properties. In the dialog that pops up select the Key check box and save the record. Then click on the Build bar item, select Current Definition, click Key, select Execute SQL Now, and click the Build button.

Checking Space and Resources

Before running the `SWAP_BASE_LANGUAGE` command, be sure that your database has sufficient space and resources to perform the swap process. Although the swap base language process commits after each table is successfully swapped, you need sufficient log space, or rollback space in the case of Oracle databases, to hold the contents of the largest table to be swapped.

On a newly installed PeopleSoft database, the `PSPNLFIELD` table is typically the largest table to be swapped, and this can be used as a benchmark for sizing your log space. Plan on at least 50 to 75 MB log or rollback segment space before running the swap process.

Running the `SWAP_BASE_LANGUAGE` Data Mover Command

Once you are ready to swap the base language, start PeopleSoft Data Mover in non-bootstrap mode by logging in using a regular PeopleSoft user ID (not the access ID). Once PeopleSoft Data Mover begins, type and run the `SWAP_BASE_LANGUAGE` command:

```
SWAP_BASE_LANGUAGE <target language>;
```

PeopleSoft Data Mover provides feedback during the swap process, including the name of the record currently being swapped and the number of records that remain to be swapped.

Even when `SWPAUDIT` completes with no errors, the swap base language process may fail. Typically, environmental issues, such as lack of database space, log space, or rollback segment space, causes any errors during this stage of the process. If a failure occurs, note the database-specific error message issued and take the appropriate action according to your database platform documentation.

PeopleSoft Data Mover stops when errors are encountered. Once you have corrected the problems that caused the failure in the swap base language process, you can restart the process without having to restore your database or remember where the first error occurred. To restart the swap process, rerun the `SWAP_BASE_LANGUAGE` command. PeopleSoft Data Mover recognizes the tables whose data has already been swapped and does not attempt to swap the data in those tables again; it will report that zero rows were swapped for those tables.

It is safe to rerun this command as many times as needed, correcting errors between runs, until the log file reports no errors.

In some situations, you may want to swap a specific record. Typically, you would swap a single record only if errors occurred during the swap base language process and you want to verify that the swap will succeed without having to re-swap all the records in the database or trace or troubleshoot the swap process.

To swap a specific table, use the following PeopleSoft Data Mover commands:

```
SET BASE_LANGUAGE <target language>;
SWAP_BASE_LANGUAGE <rename>;
```

Note. You should swap individual tables only when there has been an error with system-wide swapping.

If you want PeopleSoft Data Mover to not stop when errors are encountered and continue the swap process, set the following:

```
SET IGNORE_ERRORS;
SWAP_BASE_LANGUAGE <LANGUAGE_CD>;
```

If you chose to ignore errors during the swap, after the swap process is complete carefully review the log file to ensure that no errors occurred. If errors occur, fix the errors and then re-swap each table using the `SWAP_BASE_LANGUAGE` command in individual table mode for each table that failed.

See Also

PeopleTools 8.45 Installation Guide for Your Database Platform

Creating Related Language Tables

PeopleSoft applications provide related language tables for most edit tables that you are likely to translate, so in most cases, you won't have to create your own related language tables. However, if during a customization you create a new table that requires translation, you must create your own related language table.

The following procedure assumes that you know how to design and build records in PeopleSoft Application Designer.

Note. If there is a field defined as a Long (DESCRLONG) on the related language table, then you must ensure that the table is in a tablespace dedicated to tables containing LONG/IMAGE columns (for example, PSIMAGE tablespace for PeopleTools-owned tables). This is needed to support DB2 UDB for OS/390 and z/OS restrictions on bufferpool sizes.

To create a related language table:

1. Check the structure of the base table record, noting its key fields, and make sure that it has fields that can be language-sensitive, such as DESCR and DESCRSHORT.

If the base table has no fields that are language sensitive, then it does not need a related language record.

2. Design the related language record.

By convention, the related language record name should reflect its relationship to the base table and end with *LANG*. For example, if the base table is *MY_NEW_TBL*, the related language table might be named *MY_NEW_LANG*.

The related language record must:

- Have all the same key fields as the base table.
- Have an additional key field, which must be *LANGUAGE_CD*.
- Have the language-sensitive fields from the base table.
- Not contain any fields that are not present on the base table except the *LANGUAGE_CD* key field.
- Not contain any non-key, non-language-sensitive fields from the base table.

3. Build the related language table.

An easy way to build the related language table is to copy the base record definition using the Save As option on the File menu. Once you save the record with a new name, you can remove the fields that are not required on the related language table and add the *LANGUAGE_CD* key. By using this approach, you avoid having to remember the key structure and column names of the base table.

Note. When using Save As to create the related language record, you do not need to also save the PeopleCode that is associated with the base record. PeopleCode programs on related language records are not executed; therefore, they are redundant and may be misleading. For this reason, you should not maintain PeopleCode programs on related language records.

It is also recommended that you remove any search key and alt search key attributes from the related language table, as these do not need to be translated and can hinder performance.

4. Associate the related language table with the base table record.

Open the Record Properties dialog box for the base table. On the Use tab, select the related language record in the Related Language Record field.

Click OK to close the Record Properties dialog box.

5. Save the record.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Application Designer, “Creating Record Definitions”

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Application Designer, “Administering Data”

Creating Related Language Views

This section provides an overview of related language views and discusses how to:

- Create views for one base table and one related language table.
- Create views for two base tables and one related language table.
- Create views for two base tables and two related language tables.

Understanding Related Language Views

Just as records with language-sensitive fields require related language records, views with language-sensitive fields require related language views. This is necessary because the view is created at the database-level outside of PeopleTools and cannot recognize when one or more of the tables it selects from has a related language table. If you create a view over a table that has a related language record and don't create the corresponding related language view, any data selected from your view will appear only in the base language of the database.

Related language views work the same way as related language records:

- You can create a related language view for any view that selects language-sensitive fields from a table with a related language record. Anytime you create a view over a table that has a related language record, you typically also need a related language view.
- The related language view consists of all the key fields from the base view, a language code, and the language-sensitive fields.
- You associate the base view and the related language view in the Record Properties dialog box of the base view.
- When a user logs on in a non-base language, the PeopleSoft system retrieves the data with the appropriate language code from the related language view.

- It's best to use the same naming convention for related language views as you do for related language records: append a *LANG* suffix to the view name.

Related language views have one additional issue: the *SELECT* statement from the original view must be modified to select any language-sensitive fields from the appropriate related language tables.

The *SELECT* statements behind the views vary in complexity, depending on how many tables are involved and how many of those tables have related language records.

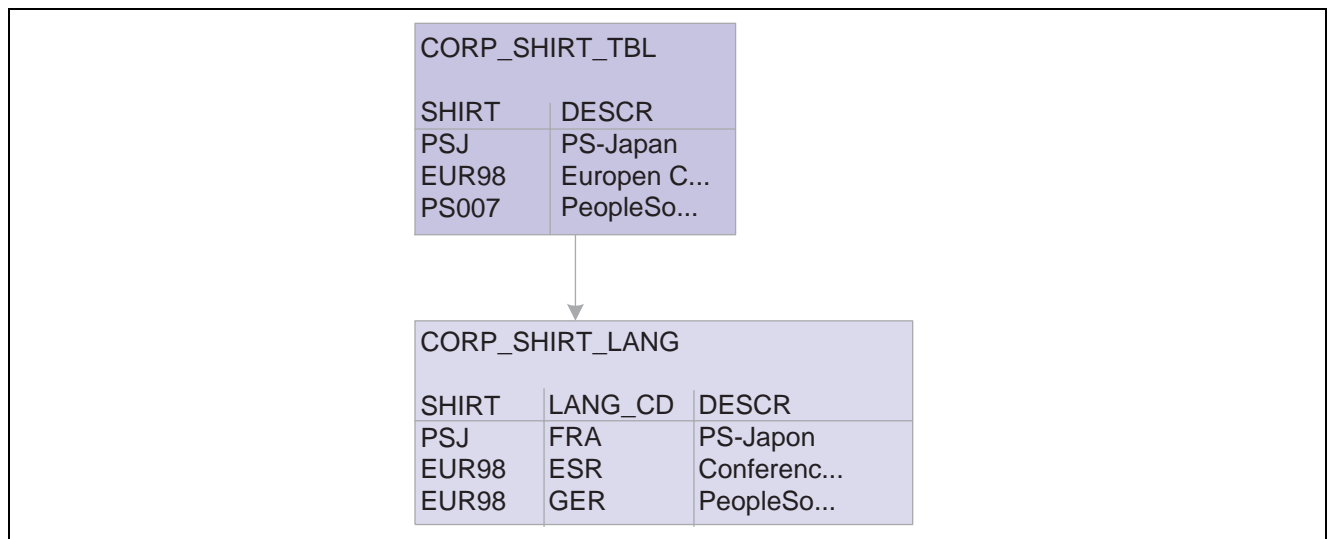
The following sections discuss how to construct the *SELECT* statements for the following language-sensitive views:

- One base table, one related language table.
- Two base tables, one related language table.
- Two base tables, two related language tables.

Creating Views for One Base Table and One Related Language Table

When you have a view that selects data from a single table, the select statement for the language-sensitive view is straightforward: select the data from the related language table, making sure to also select *LANGUAGE_CD*.

For example, this figure shows a view on the *CORP_SHIRT_TBL*, a table keyed by *SHIRT* and with one language-sensitive field (*DESCR*):



One base table, one related language table

Let's assume you must create a view that selects only the rows where the *SHIRT* key field begins with the letter *P*.

Following is the *SELECT* statement for the base view:

```
SELECT SHIRT, DESCR FROM PS_CORP_SHIRT_TBL WHERE SHIRT LIKE 'P%'
```

Following is the select statement for the language-sensitive view:

```
SELECT SHIRT, DESCR, LANGUAGE_CD FROM PS_CORP_SHIRT_LANG WHERE SHIRT LIKE 'P%'
```

Because this view is simple and selects only columns that exist both in the base table and in the related language table, the related language view is straightforward. It differs from the base view in only two ways:

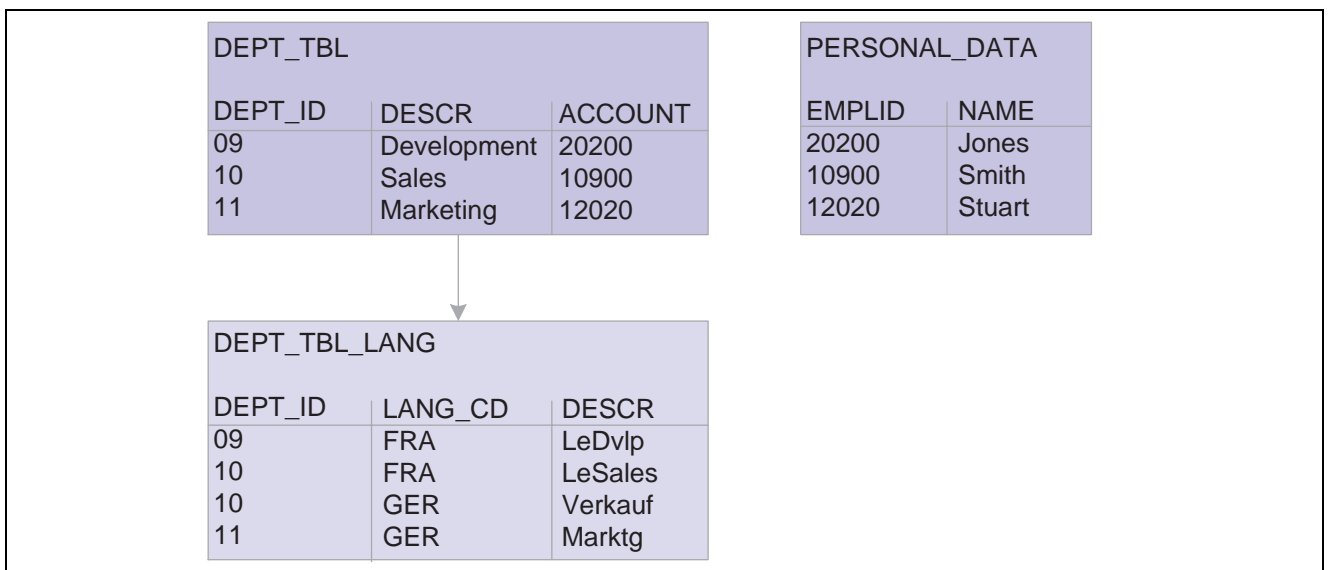
- The name of the table from which it selects.
- The addition of the LANGUAGE_CODE column.

If the base view also selected non-key, non-language-sensitive columns from the base table, there would be no need to select those columns in the related language view because non-language-sensitive fields don't need to be included in related language records.

Creating Views for Two Base Tables and One Related Language Table

When you join two tables, one of which has a related language record, the SELECT statement becomes only slightly more complex.

For example, this figure shows a view that joins DEPT_TBL, which has a related language record, with PERSONAL_DATA, which does not:



Two base tables, one related language table

The following SELECT statement is for the base view that selects the department ID and description and the name of the manager for each department:

```
SELECT A.DEPT_ID, A.DESCR, B.NAME
FROM PS_DEPT_TBL A,
     PS_PERSONAL_DATA B
WHERE A.MANAGER = B.EMPLID
```

The following SELECT statement is for the language-sensitive view:

```
SELECT A.DEPT_ID, A.DESCR, C.NAME, A.LANGUAGE_CD
FROM PS_DEPT_TBL_LANG A,
     PS_DEPT_TBL B,
     PS_PERSONAL_DATA C
WHERE A.DEPT_ID = B.DEPT_ID
AND B.MANAGER = C.EMPLID
```

In this case, the related language view can't get all the information from the related language table. This is because the MANAGER field, which is used to get the manager's name from PS_PERSONAL_DATA, is not language-sensitive and is therefore not part of the related language table.

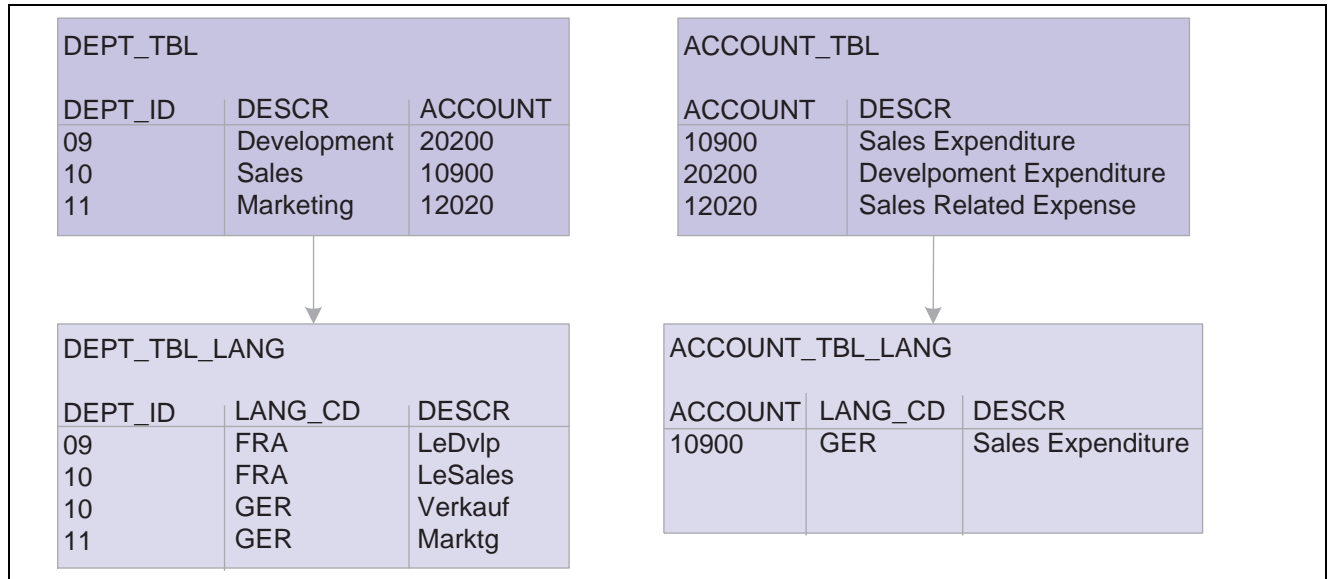
To retrieve the MANAGER field, PS_DEPT_TBL_LANG needs to be joined to PS_DEPT_TBL.

Because the PERSONAL_DATA record does not have a related language table, no special logic is required for fetching the manager's name because it is the same in all languages.

Creating Views for Two Base Tables and Two Related Language Tables

When you join two tables that both have related language records, the SELECT statement becomes complex.

For example, this figure shows a view that joins DEPT_TBL with ACCOUNT_TBL:



Two base tables, two related language tables

In this example, the base view joins DEPT_TBL with ACCOUNT_TBL to retrieve the description of the account. The following SELECT statement is for the base view:

```
SELECT A.DEPT_ID, A.DESCR, B.ACCOUNT, B.DESCR
FROM PS_DEPT_TBL A,
     PS_ACCOUNT_TBL B
WHERE A.ACCOUNT = B.ACCOUNT
```

Creating the related language view is difficult because both tables referenced in the base view have related language records. Because the related language architecture does not require that all rows have translations, creating the related language view is not as simple as joining DEPT_TBL_LANG with ACCOUNT_TBL_LANG.

The related language view needs to take into account the following scenarios:

- A translated department may reference a translated account.
- A translated department may reference an untranslated account.
- An untranslated department may reference a translated account.

In this scenario, there is no row in DEPT_TBL_LANG, but there is a row in ACCOUNT_TBL_LANG. If you join DEPT_TBL_LANG with ACCOUNT_TBL_LANG in the view, no translation is retrieved by the related language view because the department doesn't have a translation—even though the account did.

- An untranslated department may reference an untranslated account.

The related language view doesn't need to address this situation because no translation exists in this case. It's correct for the related language view to return no rows.

To address these scenarios, the SQL statement for the related language view must include logic for the first three scenarios—those where translations exist. Following is the SELECT statement for the language-sensitive view:

```
SELECT A.DEPT_ID, A.DESCR, C.ACCOUNT, C.DESCR, A.LANGUAGE_CD
FROM PS_DEPT_TBL_LANG A,
     PS_DEPT_TBL B,
     PS_ACCOUNT_TBL_LANG C
WHERE A.DEPT_ID = B.DEPT_ID
     AND B.ACCOUNT = C.ACCOUNT
     AND A.LANGUAGE_CD = C.LANGUAGE_CD
UNION
SELECT E.DEPT_ID, E.DESCR, G.ACCOUNT, G.DESCR, E.LANGUAGE_CD
FROM PS_DEPT_TBL_LANG E,
     PS_DEPT_TBL F,
     PS_ACCOUNT_TBL G
WHERE E.DEPT_ID = F.DEPT_ID
     AND F.ACCOUNT = G.ACCOUNT
     AND NOT EXISTS (SELECT 'X'
                    FROM PS_ACCOUNT_TBL_LANG H
                    WHERE H.ACCOUNT = G.ACCOUNT
                         AND H.LANGUAGE_CD = E.LANGUAGE_CD)
UNION
SELECT I.DEPT_ID, I.DESCR, J.ACCOUNT, J.DESCR, J.LANGUAGE_CD
FROM PS_DEPT_TBL I,
     PS_ACCOUNT_TBL_LANG J
WHERE I.ACCOUNT = J.ACCOUNT
     AND NOT EXISTS (SELECT 'X'
                    FROM PS_DEPT_TBL_LANG K
                    WHERE K.DEPT_ID = I.DEPT_ID
                         AND K.LANGUAGE_CD = J.LANGUAGE_CD)
```

This view is really three separate SQL statements whose output is concatenated using the SQL UNION operator. Each SELECT statement in the view addresses one of the first three scenarios previously described. Let's examine this view, statement by statement.

Statement One

This following statement addresses scenario one. It retrieves the rows where translations for both the department and the account exist.

```
SELECT A.DEPT_ID, A.DESCR, C.ACCOUNT, C.DESCR, A.LANGUAGE_CD
FROM PS_DEPT_TBL_LANG A,
     PS_DEPT_TBL B,
     PS_ACCOUNT_TBL_LANG C
WHERE A.DEPT_ID = B.DEPT_ID
     AND B.ACCOUNT = C.ACCOUNT
     AND A.LANGUAGE_CD = C.LANGUAGE_CD
```

Statement Two

The following statement addresses scenario two; it addresses instances where the department translation exists, but the account translation does not exist.

The sub-SELECT statement is required in order to prevent this statement from retrieving records that have already been selected by the statements that address scenario one, where a translated account may exist.

```
SELECT E.DEPT_ID, E.DESCR, G.ACCOUNT, G.DESCR, E.LANGUAGE_CD
FROM PS_DEPT_TBL_LANG E,
     PS_DEPT_TBL F,
     PS_ACCOUNT_TBL G
WHERE E.DEPT_ID = F.DEPT_ID
     AND F.ACCOUNT = G.ACCOUNT
     AND NOT EXISTS (SELECT 'X'
                    FROM PS_ACCOUNT_TBL_LANG H
                    WHERE H.ACCOUNT = G.ACCOUNT
                         AND H.LANGUAGE_CD = E.LANGUAGE_CD)
```

Statement Three

The following SELECT statement addresses scenario three, where translations exist for the account but not the department. Again, the sub-SELECT statement is needed to avoid returning rows that have already been selected by the other SELECT statements in the SQL statement where a translated department exists.

```
SELECT I.DEPT_ID, I.DESCR, J.ACCOUNT, J.DESCR, J.LANGUAGE_CD
FROM PS_DEPT_TBL I,
     PS_ACCOUNT_TBL_LANG J
WHERE I.ACCOUNT = J.ACCOUNT
     AND NOT EXISTS (SELECT 'X'
                    FROM PS_DEPT_TBL_LANG K
                    WHERE K.DEPT_ID = I.DEPT_ID
                         AND K.LANGUAGE_CD = J.LANGUAGE_CD)
```

CHAPTER 11

Working With Language-Sensitive Application Data

This chapter provides an overview of data editing in related language tables and base tables and discusses how to edit data in multiple languages.

See Also

[Chapter 15, “Translating Application Definitions,” page 217](#)

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” page 101](#)

[Chapter 10, “Using Related Language Tables,” page 149](#)

Understanding Data Editing in Related Language Tables and Base Tables

When a user changes data on a language-sensitive page, the system responds by changing the contents of the base table, the related language table, or both, depending on the language of the user and which field was modified. The following table explains at the database table level the actions that are performed when you work with language-sensitive tables.

Action	Signon Language	Result
Translate a language-sensitive field. (The base language field has never been translated.)	Non-base	Adds a new row to the related language table that is keyed to the current sign-in language.
Edit a language-sensitive field.	Base	Changes a field in the base table without affecting the related language table.
Edit a language-sensitive field.	Non-base	Changes a field in a row of the related language table that is keyed to the current sign-in language. Doesn't affect the base table.
Edit a non-language-sensitive field.	Base	Changes a field in the base table without affecting the related language table.

Action	Signon Language	Result
Edit a non-language-sensitive field.	Non-base	Changes a field in the base table without affecting the related language table. As the related language table contains only language-sensitive fields, no changes to this table are necessary.
Add a row (new key).	Base	Adds a new row to the base table without changing the related language table.
Add a row (new key).	Non-base	Adds a new row to the base table and a new row, keyed to the current sign-in language, to the related language table. This could introduce non-base language descriptions into the base language table.
Insert a row (effective-dated).	Base	Adds a new effective-dated row to both the base table and related language table. Add a new effective-dated rows to the related language table for each language for which translations exist for the object. Any updates to language-sensitive fields are applied only to the base table.
Insert a row (effective-dated).	Non-base	Adds a new effective-dated row to both the base table and the related language table for all languages for which translations exist for the object. Any updates to language-sensitive fields are applied to the current language in the related language table and to the base table.
Delete a row.	Base or non-base	Deletes the base table row and all dependent rows in the related language table.

Editing Data in Multiple Languages

This section provides an overview of multi-language entry and discusses how to:

- Enable multi-language entry.

- Enter data in multiple languages.

Understanding Multi-Language Entry

There are two methods of maintaining translations of application data.

- By logging into the system in a language other than the database’s base language and updating a field that is language-sensitive.
- By using the Multi Language Entry option.

This section provides an overview of the Multi Language Entry option.

Using the Multi Language Entry option provides significant benefits over simply changing the session language and overwriting existing base language descriptions with their translations. It allows you to:

- Enter or edit data in multiple languages during the same session without changing the sign-in language.
- View all translations of a row of data in a single session, which may be important for multilingual implementations that need to describe data in a different language.
- Easily recognize which fields on a page are language-sensitive and which are language-neutral.

Multi Language Entry makes it more intuitive for a multilingual user, such as a Canadian operator who is proficient in both French and English, to maintain data in several languages before saving a page.

Multi Language Entry is active if both of the following are true:

- The Multi Language Entry option is enabled for the user operator.
- The current page contains a record that has a related language record.

Enabling Multilanguage Entry

Multilanguage entry makes it more intuitive for a multilingual user to maintain data in several languages before saving a page.

Option Category: General Options

Personalizations			Find	First	1-5 of 5	Last
Personalization Option	Default Value	Override Value				
Accessibility Features	Accessibility features off	<input type="text" value=""/>	Explain			
Excel 97 grid download	N	<input type="text" value=""/>	Explain			
Time page held in cache	900	<input type="text" value=""/>	Explain			
Multi Language Entry	No	<input type="text" value="Yes"/>	Explain			
Spell Check Dictionary	Use session language	<input type="text" value=""/>	Explain			

Personalizations: General Options page with Multi Language Entry enabled

To enable multi language entry:

1. Select My Personalizations on the navigation menu from your homepage.

The Personalizations page appears.

2. Click the Personalize Options button on the General Options line.

The Option Category General Options page appears.

3. Change the value in the Multi Language Entry Override Value dropdown list to *Yes*.
4. Click OK.

Entering Data in Multiple Languages

Once you have enabled Multi Language entry in My Personalizations, the page display changes in two important ways.

Data Language: English ▼ [New Window](#) | [Help](#)

Generic Template Definition
Blackberry Email Responses

Template: SYSTEMDEFAULT

***Description:**

Instructional Text:

Priority:

***Sender:** **Email ID:**

Subject:

Message Text:

Below is the list of available variables for this template.
You can use template variables within your subject or message text.
The following variables can also be used:
%Date, %DateTime, %Time, %ServerTimeZone, %EmailAddress, %NotificationPriority,
%NotificationToList, %NotificationCCList

Template Variables	
*Value	*Description
%1	URL of the source transaction

Generic Template Definition page with Multi Language Entry enabled

1. A Data Language dropdown list appears at the top of each page. The Data Language dropdown list indicates in which language the language-sensitive fields on the current page are being maintained. You can switch the display of the page between different data languages and maintain translations in each, by changing the selected language in the dropdown menu.
2. Any fields that are language-sensitive (that is, they are translatable as they exist both on the base language and related language records and are not keys) appear with a pale green shaded background. This makes it very easy to determine if the changes you make to the data will take effect in the current language only, or take effect across all languages in the database.

For example, in the previous graphic, the Subject field is language-sensitive. A different translation of the subject can be entered for each language enabled in the database, but the Email ID field is not, so any changes to the Email ID will be visible to all system users.

To translate this data into another language, simply pick a new language from the Data Language dropdown list. If translations exist for the current row of data in the new language, those translations will appear. If not, the base language values will still be displayed, and you can translate these into to the new language selected.

Note. Changing the data language of a page using Multi Language Entry only changes the language of application data; it does not change the language of the user interface of the page such as field labels and help text. The language of the user interface is controlled only by your sign-in language preference.

As changing the data language does not change the labels and other user interface elements of a page, it is also useful to determine terminology in different languages. For example, if Department descriptions in your Department table have been translated into several different languages, you can view the description of a department in each language simply by navigating to the Department table page and changing the data language. Because this only changes the language of the application data and not the user interface, you can "look up" terminology in other languages without needing to be able to navigate or use a page that has been translated into that language.

To enter data in multiple languages:

1. Enable the Multi Language Entry function.

See [Chapter 11, "Working With Language-Sensitive Application Data," Enabling Multilanguage Entry, page 171](#).

2. Navigate to the page on which the data is to be maintained.

Any language-sensitive fields on the page are highlighted in green to make it easy to identify fields that permit data maintenance in multiple languages.

3. From the Languages menu, choose a target language.

The language that appears on the page remains the same; only the language-sensitive data is reloaded in the selected language.

4. Enter or edit data in any language-sensitive fields in the target language.
5. When finished editing the multilingual fields, save the page.

The system updates the base and related-language tables for the page.

PART 4

Translating PeopleSoft Applications

Chapter 12
Modifying Terminology

Chapter 13
Adding New Languages

Chapter 14
Translating PeopleTools

Chapter 15
Translating Application Definitions

CHAPTER 12

Modifying Terminology

This chapter provides an overview of terminology management, lists prerequisites, and discusses how to:

- Grant terminology search privileges.
- Search terminology.
- View and replace terminology search results.
- View and undo terminology replacement results.
- Reviewing terminology search information.
- Selecting terminology search objects.
- Make upgrade considerations.

Understanding Terminology Management

PeopleSoft provides two sets of translation and terminology modification tools:

1. Translation tools for translating application objects into languages that are not supplied by PeopleSoft or for translating your customizations.
2. A Terminology Management tool that enables you to modify installed, PeopleSoft-delivered translations to suit local needs.

This chapter discusses the Terminology Management tool. The other chapters in this part discuss the tools that are provided for translating customizations or translating objects into new languages.

See [Chapter 13, “Adding New Languages,” page 199](#).

This section discusses:

- Terminology Management overview.
- Terminology searches.
- Terminology replacement.
- Terminology replacement undo process.
- Terminology search statuses.

Terminology Management Overview

The PeopleTools Terminology Management tool enables you to streamline certain user interface language changes through the use of text search and replace processing. This is particularly relevant to the task of adapting a language that is based on another language. For example, if your Mexican users require that their Spanish user interface reflect terms specific to Mexico, you can use the Terminology Management tool to adapt the PeopleSoft-delivered Spanish translations to substitute Mexican Spanish terms where appropriate.

You can also use this tool to change PeopleSoft-provided user interface terminology to better suit your organization’s internal terminology or corporate vocabulary, even if English is your only user interface language.

The Terminology Management tool only adapts an existing installed language—it cannot create a new language based on the changes you define. If you want to preserve the existing translation and use your modifications to create a new PeopleSoft language, you must first define the new language and copy the existing translations to your new language.

Searching and replacing terms is a multistep process, with several opportunities to back out the changes. The process breaks down into the following three phases, each of which is discussed in this section:

1. Search
2. Replace
3. Undo replacement

Each phase consists of three separate steps: setting up the process, running the process, and reviewing the results. However, you don’t have to use all three processes all the time. You can use the Search process without ever invoking the Replace or Undo processes.

Terminology Searches

The first phase in terminology management is the Search phase, and features three steps.

Step	Description
Setting up the Search.	On the Define Search Criteria page, specify which languages to search, which database objects to search, and the term for which to search. To replace the term with a new term, also specify the replacement text.

Step	Description
Processing the Search.	Using PeopleSoft Process Scheduler, run the Search process.
Reviewing the results.	<p>Review the results in one of two places, depending on whether Replace processing is enabled. Replace processing can be disabled for specific searches or for specific users. For example, if you are considering replacing terminology and want to determine the potential exposure of such a change, disable the Defining a Search with Replacement option. This ensures that there is no risk of accidental text replacement.</p> <p>If Replace processing is enabled, review the results in the . If you don't want to replace the text, you can mark the process as complete.</p> <p>If Replace processing is disabled, review the results on the Search Only Results page.</p>

Terminology Replacement

The second phase in terminology management is the Replace phase, which features three steps.

Step	Description
Setting up the Replace.	<p>If your original search criteria specified replacement text, you will see both the original text and the replacement text on the Search Results/Replace pages. At this point, nothing has actually been replaced.</p> <p>Accept, reject, or modify the replacement text for each search result.</p>
Processing the Replace.	Using PeopleSoft Process Scheduler, run the Replace process.
Reviewing the Replace.	<p>Go to the Replace Results/Undo page to see the list of results, along with any replacement text. This page displays all results from the original search and indicates whether you replaced text.</p> <p>At this point, the replacement has happened, but you can still back out the changes by continuing to the undo step.</p> <p>If you don't want to undo any of the replacements, mark the process as complete.</p>

Terminology Replacement Undo Process

The third phase in terminology management is the optional Undo phase, which features three steps.

Step	Description
Setting up the Undo.	As you review your replace results on the Replace Results/Undo page, specify whether to undo any of the replacements.
Processing the Undo.	Using PeopleSoft Process Scheduler, run the Undo process.
Reviewing the Undo.	<p>Go to the Inquiry page that displays all search results to see the final results of all your searching, replacing, and undoing.</p> <p>This page displays all results from the original search, regardless of whether you replaced the term or used undo for any replacement.</p> <p>The Search and Replace is final and the record of what you did is permanent. If you ever need to back out the changes, you can use this information to research what replacements took place.</p>

Terminology Search Statuses

As you proceed through the Search and Replace process, you will notice that all of the pages display a Status field. This status is updated automatically to reflect where you are in the process.

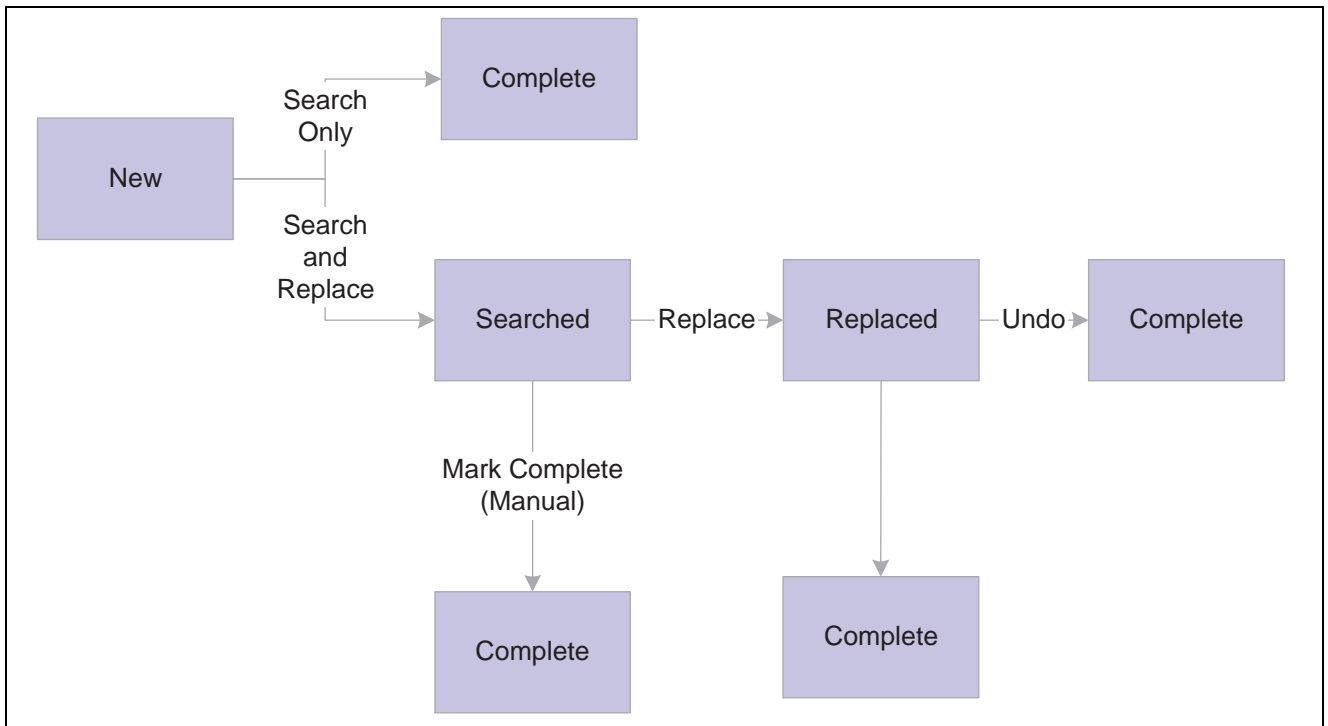
The default status for all new searches is *New*.

If you configure the search type to disable replacing (that is, if you choose the search-only option when you set up your search criteria), then no further action is possible after the search. Therefore, after the search, the status is updated to *Complete*. Otherwise, the status is updated to *Searched*.

Running the Replace process updates the status to *Replaced*; running the Undo process updates the status to *Complete*.

If you don't want to run the full Search, Replace, and Undo processes, manually mark the status *Complete* after searching or replacing. Once the status is *Complete*, you cannot change your mind. If you decide to make further terminology changes, you must recreate the search criteria using the Copy Search Criteria feature on the Search Criteria Inquiry page.

The following figure shows all the possible search statuses and their relationships:



Search statuses

Prerequisites

Before you begin modifying PeopleTools terminology, ensure that you have installed the required files and software tools.

See [Chapter 14, “Translating PeopleTools,” Prerequisites, page 209](#).

Granting Terminology Search Privileges

Use the User Search Options page to grant search and view access to your users

To set user search options:

1. Select PeopleTools, Translations, Modify Terminology, User Search Options.
2. Search for an existing user ID or add a new one.

Adding a user ID does not mean that you are creating new users; it means that you are adding a user to the list of those who have permission to use the search and replace functionality.

Add a user ID to grant access to someone who does not already have access. Search for an existing user ID to modify permissions for someone who already has access.

Use the standard search or add method to enter the user ID and access the User Defaults page.

3. Set the user’s search permissions.

Search Only Select this option to grant the user permission to search, but not to replace or undo. When a user with this level of access defines search criteria, the Search Only (No Replace) option is selected automatically.

Search, Replace and Undo Select this option to grant the user full access to all aspects of the PeopleSoft terminology management functionality.

4. Choose which searches the user can access.

Select View Other User's Activity to grant the user permission to see searches created by all users.

Leave this check box clear to deny access to all searches other than the user's own searches.

Only users with access to Search, Replace, and Undo can be granted access to other users' activity.

5. Save the page.

Searching Terminology

This section discusses how to:

- Define search criteria.
- Clone existing search criteria.
- Run the Search process.

Defining Search Criteria

The first step in a Search and Replace process is to define your search criteria. To define your search, select PeopleTools, Translations, Modify Terminology, Define Search Criteria.

Note. In addition to using the standard add method to create a new search, you can copy an existing search and then modify the definition as necessary.

Search Profile

User ID: QEDMO

Base Lang: English

Search Lang: Spanish

Identifier: ARTICULO

Base Language
 Non-Base Language
 Both

***Status:** New

Search Ready
 Search Only (No Replace)
 Search Long Fields

Search Objects

<input checked="" type="checkbox"/> Records, Fields, Xlats	<input checked="" type="checkbox"/> Menus	<input checked="" type="checkbox"/> Messages
<input checked="" type="checkbox"/> Business Processes	<input checked="" type="checkbox"/> Exclude & from Menu Search	<input type="checkbox"/> Application Messages
<input checked="" type="checkbox"/> Queries and Strings	<input checked="" type="checkbox"/> Other System Objects	<input type="checkbox"/> PeopleTools Messages
<input checked="" type="checkbox"/> Pages	<input checked="" type="checkbox"/> Application Objects	<input checked="" type="radio"/> All Messages

Search / Replacement Text Find | View All | First 1-2 of 2 Last

	Base Search Text	Non-Base Search Text	Replacement Text
ABC abc Abc	<input type="text" value="PRODUCT"/> Full Word Match <input checked="" type="checkbox"/> Match Case?:	<input type="text" value="PRODUCTO"/> Full Word Match <input checked="" type="checkbox"/> Match Case?:	<input type="text" value="ARTICULO"/> Preserve Original Case
ABC abc Abc	<input type="text" value="Product"/> Full Word Match <input checked="" type="checkbox"/> Match Case?:	<input type="text" value="Producto"/> Full Word Match <input checked="" type="checkbox"/> Match Case?:	<input type="text" value="Articulo"/> Preserve Original Case

Define Search Criteria - Search Profile page

Note. The Search Profile page displays only searches where the status is *New*. This means that you can edit the search criteria only while the status is *New*. You cannot edit the search criteria after running the search.

This section discusses how to:

- Enable Search processing.
- Choose which languages to search.
- Disable the Replace process.
- Define the search scope.
- Define search text.

Enabling Search Processing

To enable search processing, select the Search Ready box.

Choosing Languages to Search

To choose languages to search:

1. Select a search language in the Search Lang (search language) dropdown list.
The Base Lang (base language) display-only field displays the base language of the database.

The search language determines which language tables are searched. If you select the base language as your search language, you can search and replace data on the base language tables only. If you select a non-base language as your search language, your basis for searching can be dependent on either the base language, the search language, or both. However, you can still replace data only in the search language tables.

2. If the search language is different from the base language, select the basis for your search.

Select one of the following options:

Base Language	You can specify search text only for the base language tables. The system searches the base language tables, but the search returns results only when there's a match in the base language tables <i>and</i> a corresponding translation in the search language. That is, the Search process ignores occurrences of the word in the base language tables if there is no corresponding translation on the related language table.
Non-Base Language	You can specify search text only for the related language tables. The system searches the related language tables for the text that you have specified and return the results.
Both	You can specify search text for both the base language tables and the related language tables. In this case, the results returned are only those instances where the search criteria for both the base language and related language have been met.

Disabling the Replace Process

To disable the Replace process, select the Search Only (No Replace) check box.

Selecting this check box ensures that the criteria is used only for searching, not for searching and replacing. By selecting this option, you prevent the system from doing anything further after you run the search.

This check box also determines whether the status is set to *Complete* or *Searched* after you search. If you select the Search Only (No Replace) check box, then no further action is possible after the search and the status is updated to *Complete*. Otherwise, the status is updated to *Searched*.

Defining Your Search Scope

To define your search scope:

1. Specify whether to search long text fields.

Select the Search Long Fields check box to include the long description field in your searches. This setting applies to any type of object that you're searching.

2. Specify which groups of database objects to search.

Searchable text exists in many parts of the database: fields, queries, menus, and even your application data. Before you search, you must choose which types of database objects you want to search.

Use the Search Objects group box to specify which types of objects are searched. Each check box in this group box represents a group of searchable objects. Select the groups that you want to search.

Note. With certain PeopleSoft Application Designer objects, the translatable text shows up only in PeopleSoft Application Designer dialog boxes and on property sheets. For example, you see page descriptions in the PeopleSoft Application Designer Open dialog box, but end users don't see the descriptions when they look at the actual page. The important exception to this situation are fields, translate values, page text, and menus. The descriptions for these four types of objects are visible to end users.

Select one or more of these groups:

Records, Fields, Xlats	Includes record descriptions, field labels, and translate (xlat) values.
Business Processes	Includes the descriptions for activities and business process maps, and the labels of the icons on those maps.
Queries and Strings	Includes the descriptions for queries and strings.
Pages	Includes descriptions of pages and page text that does not come from a field definition or translate value.
Menus	<p>Includes all levels of navigation, including the component name.</p> <p>Some menu descriptions include an extra ampersand to designate a hot key. For example, a menu that looks like:</p> <p><i>Administer <u>W</u>orkforce</i></p> <p>is actually stored in the system as <i>Administer &Workforce</i>. Most likely you would prefer to disregard the ampersand when you search; to do so, select Exclude & from Menu Search.</p>
Other System Objects	Includes all other PeopleTools objects, such as application messages, component interfaces, and process definitions.
Other Application Objects	<p>Includes translatable application data—application tables with related language tables. There doesn't have to be a translation in your search language; the existence of the related language table is enough to qualify a table for inclusion.</p> <p>The PeopleSoft system provides a PeopleSoft Data Mover script, TSRECPOP.dms, which creates this list of translatable application tables. If you have customized the system with new translatable tables, you must run this script to recreate the table.</p>
Messages	<p>Includes both the messages and the longer explanatory text that is associated with each message.</p> <p>When searching messages, you can select the range of message to search. In the PeopleSoft system, message sets 1–999 are reserved for PeopleTools messages. Messages sets 1000–19999 are used by the applications. Selecting either the Application Messages or PeopleTools Messages check boxes narrows your search to the selected numeric range. You can also select All Messages.</p> <p>Alternatively, you can narrow your search to a specific range of messages. To do this, select the All Messages check box, and then click the prompt button to the right to open the page where you can enter the desired message set range.</p>

Defining the Search Text

To define the search text:

1. Enter the search and text for the base language, the search language or both.

There are three columns for search text. Different choices are available depending on the languages you have selected to search and whether you have allowed replacing.

Search Text - Base	This option is available if you are searching the base language or both the base and non-base languages.
Search Text - Non Base	This option is available if you are searching the non-base language or both the base language and the non-base language.
Replacement Text	This option is available unless you selected the Search Only (No Replace) check box.

For each language that you're searching, enter the text for which you want to search. The system searches only the appropriate language tables.

Note. You can enter multiple rows in each column. This is useful if you want to search for several different terms, or if you want to search for several forms of the same word. For example, if you want to search for *mouse* and *mice* in the same search, create two search rows and use one for each word.

2. Make sure that your search text is in the proper case.

Searching is case sensitive. When you want to search for all occurrences of a word, regardless of case, create multiple search rows and enter the word multiple times—once for each variation.

Use the three abc buttons to automatically put the text in the specified case.

ABC	Capitalizes the text.
abc	Lowercases the text.
Abc	Puts the text in title case (the first letter of each word is capitalized).

These buttons modify all text in the associated row. So if you are not using the same case in all three columns, take care not to inadvertently reset the case in one column when intentionally resetting the case in another.

3. Choose the matching method for the search.

Directly underneath each search text field, select one of the following matching methods to use when searching for that text:

Exact Text Match	Returns any instance where the text constitutes the entire contents of the field. For example, if you perform an exact text search on <i>Department</i> , your results won't include fields that contain <i>Department Code</i> .
Full Word Match	Returns all occurrences of the text string where the string is not embedded in another word. In this case, searching for <i>Department</i> returns occurrences of <i>Department Code</i> , but will not return occurrences of <i>Departments</i> or <i>Departmentalize</i> .
Like Text Match	Returns all occurrences of the text string

See Also

[Chapter 8, "Understanding Global Reporting and Data Analysis Tools," The PSSQR.INI and PSSQR.UNX Files, page 112](#)

[Chapter 12, "Modifying Terminology," Reviewing Search Criteria, page 195](#)

[Chapter 12, "Modifying Terminology," Modifying Searchable Objects, page 197](#)

Cloning Existing Search Criteria

If you want to set up multiple similar searches, expedite data entry by cloning an existing search. This enables you to start with most of your criteria already in place.

To clone a search definition:

1. Select PeopleTools, Translations, Modify Terminology, Search Criteria.
You are prompted to identify a search.
2. Enter the identifiers for the search definition.
Use the standard search method to access the Search Criteria page.
3. Click the Copy Search Criteria button.
You are prompted for a unique search identifier.
4. Enter a Text Search Identifier that will be the unique name for the new search.
5. Click OK.

Sample Search Criteria

Following are examples of possible searches:

- Searching base language text.

If you enter text in the Search Text - Base and Replacement Text sections, the search and replace is only on the base tables. This is a search and replace because the Search Only (No Replace) check box is clear. Because the search and base languages match, you can enter data in the Search Text - Base and Replacement Text sections of the page.

Notice that all of the check boxes are selected in the Search Objects area. That means that the system searches all of the available search objects for the word *Department*. When the search results are returned, you can replace each instance of the word *Department* with the term *Cost Center*. The text replacement occurs in the base language tables.

- Searching for base language text with existing translations.

For example, enter the word *Product* in the Search Text — Base section only. In this case the system searches for the word *Product* in the base language tables and shows all occurrences where *Product* has been translated into Spanish. You might do this as a preliminary effort, to see the different ways in which the word *Product* has been translated.

- Searching non-base language text.

For example, if you enter the word *Producto* (and select the full word match) in the Search Text - Non Base section, the system searches for all instances of the word *Producto* in the Spanish language tables. Again, you might do this type of search in order to see the number of occurrences in the database. The result shows all occurrences where English text has a translation with the word *Producto*.

- Replacing non-base language text.

For example if you enter the words *Producto*, *PRODUCTO*, and *producto* in the Search Text - Non Base section (and select the full word match for each instance), and enter the words *Articulo*, *ARTICULO*, and *articulo* in the Replacement Text section, the system searches for all instances of the word *Producto* and gives you the option to change each instance to the word *Articulo*.

Using the different cases enables you to catch all occurrences.

- Searching base language and non-base language text.

Select the Both option so that the only occurrences that are returned are those where the search criteria matches on both the base table and the non-base table. Let's say, for example, that you have translated the word *Product* in two different ways in your database: *Producto* and *Articulo*. You then realize that, for consistency, you want *Product* to always be translated to *Articulo*. This search ignores where *Product* has already been translated to *Articulo* and searches only for the instances where *Product* has been translated to *Producto*. This search also enables you to change each instance of *Producto* to *Articulo*.

Running the Search Process

Once you define and save your search criteria (and select the Search Ready check box), you are ready to run a search.

To run a search:

1. Select PeopleTools, Translations, Modify Terminology, Search.
2. Search for an existing run control ID or add a new one.

Use the standard search or add method to enter your run control ID and access the Search page.

3. Verify that the appropriate search definitions will be processed.

By default, the search includes all previously unprocessed search definitions (that is, definitions with the status *New*) where the Search Ready check box is selected. If a search appears here that you do not want to run, go back to the Search Profile page and clear the Search Ready check box. If a search is missing, go back to the Search Profile page and select the Search Ready check box.

Click the View Search Criteria button to display a read-only version of the search criteria. This is useful when you want to confirm that you're satisfied with the search criteria. Remember that running the search won't affect your data, but it can still be inconvenient to run the wrong search, as you can't go back and modify the search criteria.

4. Click Run.

The Process Scheduler Request page appears.

5. Complete the Process Scheduler Request page.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Process Scheduler*, "Submitting and Scheduling Process Requests".

6. Click OK.

The search runs. Once the Search process is complete, you can navigate to the Search Results/Replace page to see the results.

Viewing and Replacing Terminology Search Results

After a search is completed, select PeopleTools, Translations, Modify Terminology, Search Results/Replace to review the search results.

This section discusses how to:

- Replace non-message text.
- Replace message text.
- Run the Replace process.

Replacing Non-Message Text

The Replace Data page displays the database object search results, but not the Message Catalog search results. Use this page to review search results, verify replacement text, and indicate when to run the Replace process.

This section discusses how to:

- Review search results.
- Verify replacement text.
- Enable or disable replace processing.

Reviewing Search Results

To review search results:

1. Examine the search results in both the base language and the non-base language.

The search results appear in a grid where you can see details about each occurrence of the search text.

Base Text	Displays the occurrences of the text that you searched for in the base language tables.
Non-Base Text	Displays the occurrences of the text you searched for in the non-base language tables. This is the text that is replaced if you decide to run the Replace process. If you searched the base language only, there is no non-base text, and the base text is replaced.
Replacement Text, Too Long, and Replace	These columns are used to finalize the replacement text. <u>See Chapter 12, “Modifying Terminology,” Viewing and Replacing Terminology Search Results, page 188.</u>

2. Look to see which pages display the search text.

Pages	This column indicates the number of pages on which the instance of the search text appears. This number appears only for objects that are placed on pages: fields, translate values, and (hard-coded) page text.
View Text Search Details	Click this button to display details about that occurrence of the search text. View the record and field where this instance was found, as well as the keys for each occurrence of the text in the record. Also view the text from the base language table, the corresponding translation from the related language table, and the suggested replacement text.
View Text Found	Click this button to display information about which search words were found. Remember that a single set of search criteria can include searches for many terms. For example, if you searched for both <i>Codigo</i> and <i>Departamento</i> , the Text Found page shows which word or words were found in this record.

3. (Optional) Review the information about the record and field where the text was found.

Field Name	System name of the field where the text resides.
Length	Displays the length of the field. This information is useful when you plan to replace text because it tells you the maximum length of any replacement text.

Record Name of the record where the occurrence was found.

RLang Rec (related language record) Name of the corresponding related language table.

4. (Optional) Inspect the original search criteria.

View Search Criteria Click this button to display a read-only version of the search criteria. This can help you understand the search results.

Verifying Replacement Text

To verify replacement text:

1. Inspect the suggested replacement text.

Replacement Text Shows the suggested replacement text, based on your original search criteria. At this point no replacement has occurred. Remember that the replacement text replaces the non-base text unless you searched only the base language, in which case it replaces the base text.

2. If necessary, modify the replacement text.

Modify the replacement text if you're not happy with the suggested text or if the suggested text is too long for the field.

Too Long The system selects this check box when the suggested replacement text is too long for the field. This can occur when the replacement text is longer than the search text. When this happens, you must edit the replacement text (perhaps using an abbreviation) so that it fits in the field. It may help to look at the Length column to see the maximum allowable length.

Once the replacement text is an acceptable length, the system clears the Too Long check box.

You don't need to modify the text if you decide that you don't want the replacement to happen at all; the next step describes a different mechanism for disabling replacement for specific rows.

Note. When the replacement text is too long, the search text won't be replaced during the Replace process.

3. Selectively enable and disable replacement processing.

Replace Use this check box to replace some occurrences of the search text without replacing all of them.

Select the check box for the occurrences that you want to replace; clear the check box for the occurrences that you want left as they are. When you run the Replace process, only the selected rows are updated.

After you run a search, the Replace check box is automatically selected for all occurrences other than those where the suggested replacement text is too long for the field.

To clear the Replace check box in all rows, click the arrow that curves to the left.

To select the Replace check box in all rows (other than those where the replacement text is too long for the field), click the arrow that curves to the right.

Enabling or Disabling Replace Processing

To enable or disable replace processing:

1. To enable replace processing, select the Replace Ready check box.
The replace process picks up only searches where this check box is selected.
2. To permanently disable replace processing for this specific search, click the Complete button.
This button changes the status of the results to Complete. Once this happens, no further processing is possible. If you have edited messages on the Replace Messages page, those changes become permanent.
3. Save the page.

See Also

[Chapter 12, “Modifying Terminology,” Replacing Message Text, page 191](#)

Replacing Message Text

The Replace Messages page provides a focused view of all the Message Catalog entries that satisfy the search criteria. Use this page to review the messages and make replacement changes manually. You can use this feature at any time in the Search, Replace, and Undo process.

Note. You must manually update message text. If you continue to the Undo step, you must also manually back out of your changes. The automatic Replace and Undo process for other database objects does not apply to message text.

To review search results and update message text:

1. Click the Review Messages button to load the results into the page.
Once the results are loaded, you can see each message that contained the search text from your search.
2. (Optional) Inspect the original search criteria.
Click the View Search Criteria button to display a read-only version of the search criteria.
3. Review the messages.
You can see only one message at a time on this page. To review the messages, page through them and look at each one individually.
4. Edit the messages as necessary.
As you review the messages, determine whether it is appropriate to replace the text or reword the message. When you decide to make changes, edit the text directly on this page. Saving the page updates the Message Catalog.

Note. You must update message text manually. The automatic Replace process for other database objects does not apply to message text.

5. Save the page.

Running the Replace Process

After you have finalized your replacement text and settings in the Search Results/Replace page, you must run a Replace process to actually make the replacements. Running the Replace process updates the search status from *Searched* to *Replaced*.

After you run the Replace process, you have one last chance to back out any changes made by the process.

See [Chapter 12, “Modifying Terminology,” Replacing Message Text, page 191](#).

Warning! Once you change the search status and update the search status, you can no longer view this search in the Search Result/Update pages. Be sure that you are finished with *both* pages, that is, you have finished manually editing your messages, before you change the status.

To run the Replace process:

1. Select PeopleTools, Translations, Modify Terminology, Replace.
2. Search for an existing run control ID or add a new one.

Use the standard search or add method to enter your run control ID and access the Replace page.

3. Verify that the appropriate search definitions will be processed.

By default, the Replace includes all search definitions that have the status *Searched* and have the Replace Ready check box selected. If a search appears here that you do not want to run, go back to the Search Results/Replace page and clear the Replace Ready check box. If a search is missing, go back to the Search Results/Replace page and select the Replace Ready check box.

Click the View Search Criteria button to display a read-only version of the search criteria. This is useful when you want to confirm which set of criteria you’re processing.

4. Click the Run button.

The Process Scheduler Request page appears.

5. Complete the Process Scheduler Request page.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Process Scheduler*, “Submitting and Scheduling Process Requests”.

6. Click OK.

The Replace process runs. Once the replace is complete, you can navigate to the Replace Results/Undo page to view the results.

Viewing and Undoing Terminology Replacement Results

This section discusses how to:

- Undo data replacements.
- Undo text replacements.
- Run the Undo process.

Undoing Data Replacements

After you run the Replace process, select PeopleTools, Translations, Modify Terminology, Replace Results/Undo Replace to review your results and decide whether to undo any of the replacements.

This section discusses how to:

- Replace replacement text.
- Enable or disable undo processing.

Reviewing Replacement Text

To review replacement text:

1. Review the results of the Replace process.

The Replace Results/Undo page shows the same information as the Replace Data page.

The Base Text and, if applicable, the Non-Base Text show the original text.

You can see how many pages display the text, and you can click the View Text Search Details button to see details about these pages.

The Replacement Text field shows what was—or would have been—substituted for the original search text.

The Replaced check box indicates whether the original search text was replaced. At this point, the original text has been replaced, but the change has not been finalized.

The Field Name, Length, Record and RLang Rec columns provide details about where the text is stored in the database.

2. Selectively enable and disable undo processing.

You can undo some of the replacements without undoing all of them. Control which occurrences are replaced using the Undo check boxes.

Select Undo for those rows where you want to back out of your changes. Leave the Undo check box clear for the rows where you're satisfied with the replacement. When you run the Undo process, only the selected rows are affected.

3. Save the page.

Enabling or Disabling Undo Processing

To enable or disable undo processing:

1. To enable replace processing, select the Undo Ready check box.

The Undo process picks up only the rows where this check box is selected.

2. To permanently disable undo processing for this search, click the Complete button.

This button changes the status of the results to *Complete*. Once this happens, no further processing is possible. Any replacements that have been made, including changes to messages, are now finalized.

See [Chapter 12, “Modifying Terminology,” Replacing Message Text, page 191](#).

3. Save the page.

Undoing Text Replacements

The Undo Messages page displays the search results in the Message Catalog. If you edited the messages during the replace step, the text reflects those changes.

The Undo Messages page is identical to the Replace Messages page, and you can review and update messages using the same procedures.

To access the Undo Messages page, select PeopleTools, Translations, Modify Terminology, Undo.

Note. Just as you have to replace message text manually, so must you back out of any changes manually. The automatic replace and Undo process for other database objects does not apply to message text.

See Also

[Chapter 12, “Modifying Terminology,” Replacing Message Text, page 191](#)

Running the Undo Process

To run the Undo process:

1. Select PeopleTools, Translations, Modify Terminology, Undo.

2. Search for an existing run control ID or add a new one.

Use the standard search or add method to enter your run control ID and access the Undo page.

3. Verify that the appropriate search definitions will be processed.

By default, the undo includes all search definitions that have the status *Replaced* and the Undo Ready check box selected. If a search appears here that you do not want to run, go back to the Replace Results/Undo page and clear the Undo Ready check box. If a search is missing, go back to the Replace Results/Undo page and select the Undo Ready check box.

Click the View Search Criteria button to display a read-only version of the search criteria. This is useful when you want to confirm which set of criteria you’re processing.

4. Click the Run button.

The Process Scheduler Request page appears.

5. Complete the Process Scheduler Request page.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Process Scheduler*, “Submitting and Scheduling Process Requests”.

6. Click OK.

The Undo process runs. Once the undo is complete, navigate to the All Search Results page to view the results.

Reviewing Terminology Search Information

Three read-only pages provide you with information about searches.

This section discusses how to:

- Review search criteria.
- View results for search-only searches.
- View search results for all searches.

Reviewing Search Criteria

To see the search criteria for any search that's been defined, select PeopleTools, Translations, Modify Terminology, Search Criteria.

This page is a display-only version of the Search Profile page.

To transfer to the page that is appropriate to the current status of the search, click the View Details button. For example, if the status is *Searched*, then in order to continue the Search, Replace and Undo process, you must go to the Search Results/Replace pages. The following table explains which component you see, based on the status of the search:

Status	Component	Navigation
New	Define Search Criteria	PeopleTools, Translate, Use, Define Search Criteria
Searched	Search Results/Replace	PeopleTools, Translate, Use, Search Results/Replace
Replaced	Replace Results/Undo	PeopleTools, Translate, Use, Replace Results/Undo
Complete	All Search Results	PeopleTools, Translate, Inquire, All Search Results
Complete	Search Results Only	PeopleTools, Translate, Inquire, Search Results Only

Click the Copy Search Criteria button to clone this search definition.

Viewing Results for Search Only Searches

When a search is defined as Search Only (No Replace), the status is set to *Complete*, and prevents you from viewing the search results on the Search Results/Replace pages. Instead, you must access the result using one of the Inquiry pages.

To see the search results for a completed Search Only (No Replace) search, select PeopleTools, Translations, Modify Terminology, Search Only Results.

This page is a display-only version of the Search Results/Replace page.

See Also

[Chapter 12, "Modifying Terminology," Viewing and Replacing Terminology Search Results, page 188](#)

Viewing Search Results for All Searches

To see the search results for any search, select PeopleTools, Translations, Modify Terminology, All Search Results.

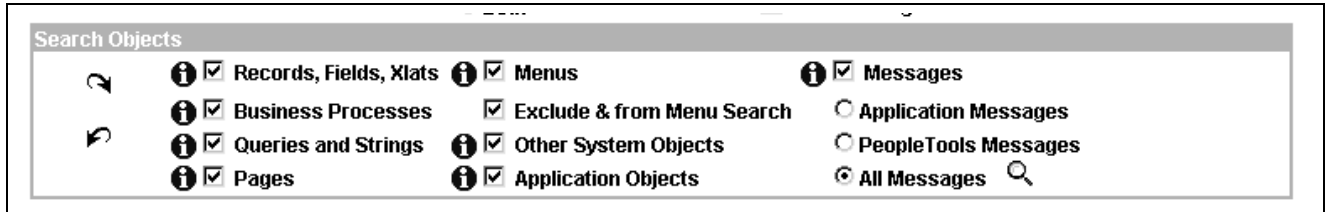
This page is a display-only version of the Replace Results/Undo page.

See Also

Chapter 12, “Modifying Terminology,” Viewing and Undoing Terminology Replacement Results, page 192

Selecting Terminology Search Objects

When you define your search, you select the types of objects to include in your search. You do this by selecting from predefined groups of object types.



Groups of object types

PeopleSoft has already defined each group by associating particular tables in each group. You can see which tables are included in each group by clicking the View Objects button, denoted by the “i” icon.

You can use the Text Search Records page to view additional information about these records. Specifically, you can identify the key fields and the searchable fields.

You can use the same page to change the association between searchable fields and the groups of object types. However, this is not advisable. The only time you should modify the object groups is when you want to add newly created tables to the Application Objects group. The PeopleSoft system provides a PeopleSoft Data Mover script to do this for you.

This section discusses how to:

- View searchable objects.
- Modify searchable objects.

Viewing Searchable Objects

Use the Text Search Records page to view additional information about the searchable records.

To view the searchable fields in any searchable record:

1. Select PeopleTools, Translations, Modify Terminology, Text Search Records.

You are prompted to identify the record to view.

You can search for records either by record (table) name or by object type. The object types are the same as the groups you use when defining a search.

2. Search for the record to view.

Use the standard search or add method to enter the user ID and access the Text Search Records page.

3. Examine the information about this record.

The Record and Related Language Record Name fields identify the record you’re viewing.

The Combined Language Table check box is selected when the base record and the related language record are the same; you cannot change this setting.

The Object Type field identifies which searchable group of objects includes this record.

Warning! Although you can modify the Object Type field on this page, it is not advisable. As delivered, records are already logically organized into appropriately named groups; changing the association causes the group names to no longer reflect the records in the group.

The Keys and the Searchable Text Fields fields display additional information about the record. All translatable fields (that is, the non-key fields in the related language record) are searchable.

Modifying Searchable Objects

If you create additional related language tables for delivered tables or develop a new translatable structure, you can make those tables searchable by adding them to the Application Objects group. To do this, run the PeopleSoft Data Mover script TSRECPOP.dms. This script refreshes the Application Objects group so that it includes all application tables that have related language tables.

Making Upgrade Considerations

Be aware that terminology updates are registered as changes to the affected database objects. This means that when you upgrade to a new PeopleSoft release, your object definitions are out of sync with those in the new PeopleSoft database.

This means that your upgrade reports, which identify changed objects, will include all the objects that have new or changed terms. Depending on the extent of your terminology changes, this can significantly impact the amount of time you spend analyzing the differences.

If you have implemented the PeopleSoft system with no customizations, you can deal with this issue by accepting the new PeopleSoft-delivered objects and then reapplying your terminology changes.

If you have customized your system and modified the delivered terminology, you may want to minimize the upgrade compare differences by doing one of the following:

- Reversing your terminology changes before the upgrade and then redoing the terminology changes after the upgrade.
- Applying your terminology changes to the PeopleSoft-delivered software before running the upgrade and compare reports.

Both of these methods cause the upgrade reports to disregard any terminology-only changes, which simplifies your analysis of these reports.

Note. Upgrading becomes more complicated and involves extra steps when you have made terminology changes. Be sure to consider this cost when you decide whether terminology changes are necessary.

CHAPTER 13

Adding New Languages

This chapter provides an overview of the addition of new language definitions to the PeopleSoft database and discusses how to add new language codes to the system.

Understanding the Addition of New Language Definitions

PeopleSoft provides translations in several languages for all end-user objects, and these translations can be licensed from PeopleSoft. However, you can maintain data in your PeopleSoft database in as many languages as required, as long as the characters needed to represent these languages exist in the character set used to create your database. Of course, if you are running a Unicode database, you can maintain data in all languages supported by the Unicode standard in a single PeopleSoft database.

As shipped, PeopleTools includes definitions primarily for the languages for which translations are provided by PeopleSoft. Definitions for some additional languages are provided in anticipation of future translations. If you plan to use additional languages in your system, you must first define these languages in PeopleTools.

There are several places where you must define new languages before PeopleTools can use and recognize them. Work with your system administrator to ensure that you complete all the necessary steps described in this section before using a new language in PeopleTools.

Note. You do not need to perform these steps for any language for which PeopleSoft has provided translations.

Adding New Language Codes to the System

To add a new language code to the system:

1. Determine the PeopleSoft and ISO codes for your language.
2. Determine the default non-Unicode character set for your language.
3. Add new translate values to the LANGUAGE_CD field.
4. Add the new language to the PSLANGUAGES table.
5. Modify the PeopleSoft Configuration Manager Windows resources to include the new language in the Configuration Manager language selection dropdown list.
6. Modify the PeopleSoft Pure Internet Architecture signon page to include a selection for the new language.
7. Add the text.properties and error.properties files to your new language.
8. Add the associated images to your new language.

See Also

PeopleTools 8.45 Installation Guide for Your Database Platform, “Loading Translations”

Determining PeopleSoft and ISO Codes for Your Language

PeopleSoft does not currently use the ISO standard language and locale identifiers in PeopleTools. Instead, PeopleTools uses a proprietary three-letter language code that is determined and maintained by PeopleSoft. When adding a new language, you must choose a three-letter language code to represent the name of your language.

Because this code is used to key several PeopleTools database objects and the PeopleTools cache file, ensure that the language code contains only three uppercase US-ASCII letters—no numbers, spaces or accented characters.

The existing PeopleSoft language codes that have been allocated are as follows:

Code	Language
ENG	English
CFR	Canadian French
DAN	Danish
DUT	Dutch
ESP	Spanish
FRA	French
GER	German
GRK	Greek
JPN	Japanese
KOR	Korean
MAY	Malay
POL	Polish
POR	Portuguese
SVE	Swedish

Code	Language
THA	Thai
ZHT	Traditional Chinese
ZHS	Simplified Chinese

In our example, we choose *NRS* for Norwegian (the *NRS* code is derived from the Norwegian name for the language, *Norsk*) as our additional language.

Once you have chosen a three-letter code for your new language, you must determine which ISO locale corresponds to that language. ISO locales are comprised of two components, a language identifier and a territory identifier, separated by an underscore or a hyphen. Typically, the language component is in lowercase letters, and the territory identifier is in uppercase letters. For example, *en_US* is the ISO locale for U.S. English, and *de_CH* is the ISO locale for Swiss German.

The language component of the locale is the two-letter language code defined by the ISO 639 standard. The territory component of the locale is the two-letter country code defined by the ISO 3166 standard.

The territory portion of the ISO locale can be omitted if you want to indicate only a language and not a country-specific derivation of that language. In our example, we plan to add the Norwegian language, so the appropriate ISO locale is *no*. If we were adding a specific dialect of Norwegian, we would use a specific ISO locale, such as *no_NO*. But as long as we're adding the generic Norwegian language, we can use *no* as the complete ISO locale code.

In most situations, you can use only the language portion of the ISO locale. However, the two-component locale is important if you are planning to add derivations of existing languages, such as Mexican Spanish (*es_MX*) or Australian English (*en_AU*). You can easily create those language derivations from a related language using the PeopleTools Terminology Management tool.

Determining Appropriate Non-Unicode Character Sets

Although the majority of the PeopleSoft system runs in Unicode, some operations, such as file system operations on UNIX/Linux platforms and integration to some third-party products, cannot use Unicode characters. For each language in your database, you must determine which non-Unicode character set should be used when Unicode operations cannot be performed. The following table lists some of the non-Unicode character sets that are supported by PeopleTools. The complete list is in the *PSCHARSETS* table in your PeopleTools database. For each new language you add, select the appropriate non-Unicode character set from this table.

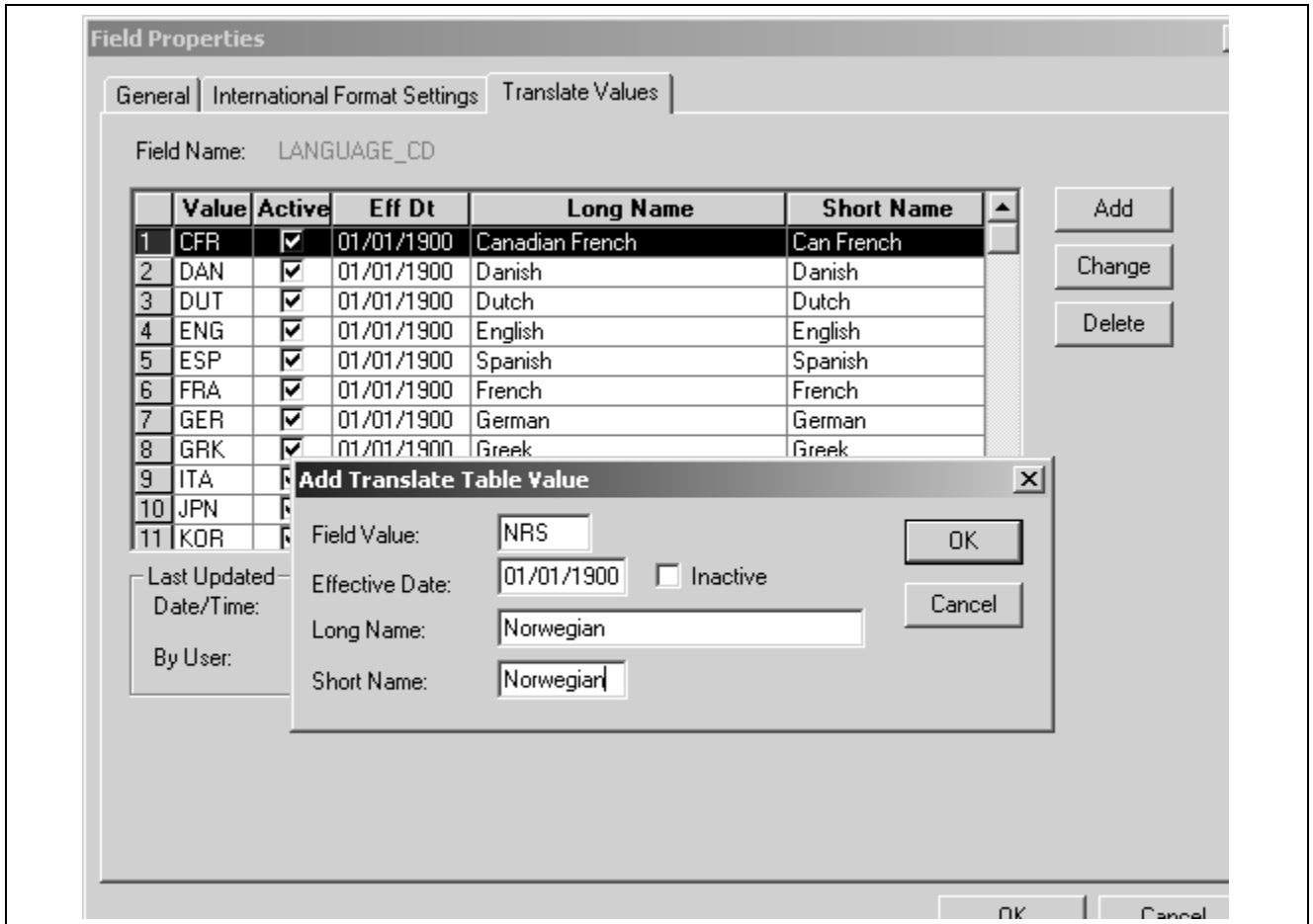
If you are not using a Unicode PeopleSoft database, the character set you select must be the character set you used to create your PeopleSoft database. For example, if you are using a DB2 UDB for OS/390 and z/OS database, you must specify the EBCDIC CCSID specified in your subsystem's *ZPARMS*.

Remember that this table is an excerpt from the complete list of character sets that are supported by PeopleTools. For the complete list, query the *PSCHARSETS* table in your SQL tool or in PeopleSoft Query.

Character Set	Description
ISO-2022-KR	ISO-2022-KR (Korean)
ISO_8859-1	ISO 8859-1 (Latin1)
ISO_8859-10	ISO 8859-10 (Latin6)
ISO_8859-11	ISO 8859-11 (Thai)
ISO_8859-14	ISO 8859-14 (Latin8)
ISO_8859-15	ISO 8859-15 (Latin9 / Latin0)
ISO_8859-2	ISO 8859-2 (Latin2)
ISO_8859-3	ISO 8859-3 (Latin3)
ISO_8859-4	ISO 8859-4 (Latin4)
ISO_8859-5	ISO 8859-5 (Cyrillic)
ISO_8859-6	ISO 8859-6 (Arabic)
ISO_8859-7	ISO 8859-7 (Greek)
ISO_8859-8	ISO 8859-8 (Hebrew)
ISO_8859-9	ISO 8859-9 (Latin5)
Shift_JIS	Shift-JIS (Japanese)

Adding New Translate Values to the LANGUAGE_CD Field

Once you have determined the PeopleSoft three-letter language code for your new language, you must add it to the Translate table so that other PeopleTools utilities and PeopleSoft applications can recognize it.



Adding a new language code

To add the language code to the Translate table:

1. In Application Designer, open the field named LANGUAGE_CD.
2. Select File, Definition Properties.
3. Move to the Translate Values tab and click the Add button.

The Add Translate Table Value dialog box appears.

4. Add the new three-letter language code and a description of your new language.

Use an effective date of 01/01/1900 to ensure that the language is always accessible to your applications. When adding the long and short names for your new language code, enter the name of the language as it is referred to in the base language of your database. Once you have defined the translate value for your language code, you can translate it into each language.

5. Click Save or select File, Save.

In our example, the database's base language is English, so we use *Norwegian* as both the long and short name of our new language. Once we have finished defining this new language, our Norwegian translators might want to translate this new value into Norwegian as *Norsk*.

Adding New Languages to the PSLANGUAGES Table

You can update and add the languages used by the PeopleSoft system to the PSLANGUAGES table using the Manage Installed Languages page. To access this page, select PeopleTools, Utilities, International, Languages.

Use the Manage Installed Languages page to specify:

- Languages in use.

The PSLANGUAGES table is used by many PeopleTools utilities to determine which languages defined in the database are actually in use, rather than being defined but unused.

- Default non-Unicode character sets.

The PSLANGUAGES table also provides a mapping to a default non-Unicode character set that is used to represent that language when a Unicode representation is not possible, for example, during some file system operations.

- Verity locale for search indexing.
- Spell check languages for language codes.

You also use the PSLANGUAGES table to select the language of the spell check dictionary that is associated with a given language code. The spell check language that you select for a language code is the default spell check dictionary that is associated with that language when the user signs on to the system in that language. The user can override the default spell check dictionary through the My Personalizations option on the homepage.

See *Enterprise PeopleTools 8.45 PeopleBook: Using PeopleSoft Applications*.

- Character sets for encoding Comma Separated-Formatted (CSV) query results.

The PeopleSoft system downloads query results to a file in comma separated value format (CSV). You can use the PSLANGUAGES table to select a charset value for a given language code to encode the results.

Language								Customize	Find	First	1-18 of 18	Last
*Language Code	Enabled	*ISO Locale	*Default Character Set	*Verity Locale Mapping	Spell Check Language	*Windows Character Set	*Verity Character Set					
1 CFR	<input checked="" type="checkbox"/>	fr-ca	ISO_8859-1	frenchx	French	CP1252	CP1252					
2 DAN	<input checked="" type="checkbox"/>	da	ISO_8859-1	danishx	Danish	CP1252	CP1252					
3 DUT	<input checked="" type="checkbox"/>	nl	ISO_8859-1	dutchx	Dutch	CP1252	CP1252					
4 ENG	<input checked="" type="checkbox"/>	en	ISO_8859-1	englishx	US and UK English	CP1252	CP1252					
5 ESP	<input checked="" type="checkbox"/>	es	ISO_8859-1	spanishx	Spanish	CP1252	CP1252					
6 FRA	<input checked="" type="checkbox"/>	fr	ISO_8859-1	frenchx	French	CP1252	CP1252					
7 GER	<input checked="" type="checkbox"/>	de	ISO_8859-1	germanx	German (new)	CP1252	CP1252					
8 GRK	<input checked="" type="checkbox"/>	el	ISO_8859-7	englishx	Greek	CP1253	CP1253					
9 ITA	<input checked="" type="checkbox"/>	it	ISO_8859-1	italianx	Italian	CP1252	CP1252					
10 JPN	<input checked="" type="checkbox"/>	ja	Shift_JIS	japanb	US and UK English	CP932	Shift_JIS					
11 KOR	<input checked="" type="checkbox"/>	ko	CP949	koreab	US and UK English	CP949	CP949					
12 MAY	<input checked="" type="checkbox"/>	rms	ISO_8859-1	englishx	US and UK English	CP1252	CP1252					
13 POL	<input checked="" type="checkbox"/>	pl	ISO_8859-2	polish	Polish	CP1250	CP1250					
14 POR	<input checked="" type="checkbox"/>	pt	ISO_8859-1	portugx	Portuguese (Brazilian)	CP1252	CP1252					
15 SVE	<input checked="" type="checkbox"/>	sv	ISO_8859-1	swedishx	Swedish	CP1252	CP1252					
16 THA	<input checked="" type="checkbox"/>	th	ISO_8859-11	uni	US and UK English	CP874	UTF8					
17 ZHS	<input checked="" type="checkbox"/>	zh-cn	GB2312	simpcb	US and UK English	CP936	GB2312					
18 ZHT	<input checked="" type="checkbox"/>	zh-tw	Big5	tradcb	US and UK English	CP950	Big5					

Manage Installed Languages page

Language Code	Enter the PeopleSoft three-letter language code from the PSXLATITEM table.
Enabled	Select the check box to indicate the language that is in use. When this check box is selected, login with the language is allowed. Normally, this check box is selected automatically during the process of loading translations into the database.
ISO Locale	Use the lookup to select the ISO locale code from the PSLOCALEDEFN table. The code consists of an ISO 639 language code, optionally followed by an ISO 3166 country code.
Default Character Set	Use the lookup to select the character set from the PSCHARSETS table. The default character set determines the default encoding for input and output files.
Verity Locale Mapping	Use the drop-down list to select which Verity locale should be used when indexing data for search in this language. If the language is not in the drop-down list for Verity Locale Mapping, select <i>englishx</i> . This mapping determines the locale to use for building search collections and searching for data.
Spell Check Language	Use the drop-down list to select the spell check language from the PSXLATITEM table. This enables you to select the language of the spell check dictionary that is associated with a given language code.
Windows Character Set	Use the Lookup option to select a Microsoft codepage to use to encode comma separated value (CSV) formatted query results for the corresponding language code. The charset must be valid for Java encoding and mime-type, or downloads will fail with an error. The default value for all language codes is <i>CP1252</i> .
Verity Character Set	Select the character set for the Verity search engine to use for its internal encoding in the given language. Normally you should not modify the value in this field.

Note. You can override the information specified on the Manage Installed Languages page using the Look and Feel tab on the Web Configuration page.

See Also

Enterprise PeopleTools 8.45 PeopleBook: Internet Technology

Modifying Configuration Manager Windows Resources

PeopleSoft Configuration Manager provides a list of languages that are available to users for their PeopleTools Windows client and Application Designer sessions. This list is embedded in the Windows Resource File for PeopleSoft Configuration Manager itself.

To add your new language to the list of languages in PeopleSoft Configuration Manager:

1. In \SRC\RES\ENG\PSCFG, edit the file pscfg.rc using a text editor.

This file stores the list of languages used in the Display tab of the PeopleSoft Configuration Manager. These languages are stored as strings with the name `IDS_LANGx`, where *x* is a sequential number. To add your new language, find the next available `IDS_LANG` entry and change the text to match your new language using the format *LanguageCode – Description*. Then, update the entry `IDS_NUMLANGUAGES` to specify the new highest `IDS_LANG` entry that you used.

File Format

Properties files are saved in a special format. US-ASCII Unicode characters must be represented by a Java-Style Unicode Escape Sequence in the format `\udddd`, where `dddd` represents the hexadecimal value of the Unicode character.

native2ascii

The `native2ascii` utility, a part of the Sun Microsystems JDK (Java Development Kit), can be used to transform your properties files into the correct format after it has been translated.

For example, to convert a non-European `text.properties` file, you will need to run the following:

```
native2ascii <sourcefile> <destination file>
```

Note. Use the destination properties files that have the Java-style encoding.

Note. For more information about the `native2ascii` utility, refer to the Sun Microsystems online documentation.

Adding Associated Images in the New Language

To add the new images in the new language, do the following:

1. Create a new directory named after your 3-letter language code in the `PORTAL\ps\images` directory.
2. Copy the following three files from the `images` directory into the `\images\<language code>` directory:
 - `customerEmployeeSupplier.gif`
 - `PT_PROCESSING.gif`
 - `tagLine.gif`
3. Change the filename to `_<language code>` before the `.gif` extension.

For example, your `\images\NRS` directory would contain these files: `customerEmployeeSupplier_NRS.gif`, `PT_PROCESSING_NRS.gif`, and `tagLine_NRS.gif`.

CHAPTER 14

Translating PeopleTools

This chapter provides an overview of PeopleTools translation, lists prerequisites, and discusses how to:

- Use alternate language dynamic-link libraries (DLLs).
- Locate resource directories.
- Translate resource files.
- Translate objects.

Understanding PeopleTools Translation

The PeopleSoft system provides utilities and methods for translating all definitional *application objects* built using PeopleTools, such as pages and reports. However, there are also some *PeopleTools components* that are visible to end-users and should be translated as well. These include messages issued by the Application Server, the Windows components of PeopleTools (such as the PS/nVision designer and PeopleSoft Query) and images used for standard navigation in the PeopleSoft Pure Internet Architecture.

This chapter explains the different translatable PeopleTools components, and how you can customize the translations delivered by PeopleSoft or translate PeopleTools into new languages.

See Also

[Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” page 101](#)

Prerequisites

Before you begin modifying or creating new translations of PeopleTools, ensure that you have the required files and software tools.

This section discusses:

- Required files.
- Required tools.

To modify translations of PeopleTools or to translate PeopleTools into a new language, you must have installed the PeopleTools Language Development Kit. When you install PeopleTools, you have the option to install the PeopleTools Language Development Kit when you install the file server. If you have installed this kit, you will find a directory <PS_HOME>\SRC\RES on your file server. You can install the PeopleTools Language Development Kit at any time – if this directory does not exist on the file server where you are planning to perform the translation, rerun the PeopleTools setup program and select only this option.

PeopleSoft provides source code to the English language resource files and selected local language resources. PeopleSoft also supports header files that you need in order to produce your own alternate language DLLs. PeopleSoft also provides batch files to automate the resource compilation process.

In addition to the source code to the resource files that are delivered in the PeopleTools Language Development Kit, PeopleSoft also delivers pre-compiled alternate language DLLs for each language provided by PeopleSoft. These are installed by selecting the PeopleTools Language Pack from the PeopleTools installer. If you intend on using a translation that PeopleSoft provides but don't plan on modifying PeopleTools translations, you only need to install the language pack, not the Language Development Kit.

The tools that you need to customize Windows resources are part of Microsoft's Visual C++ compiler package. You won't have to change or compile any C++ code, but you must use the resource compiler and linker. If you are translating any resources other than strings, you must use the Microsoft Developer Studio.

See Also

PeopleTools 8.45 PeopleBook: Hardware and Software Requirements Guide

Using Alternate Language DLLs

Although the vast majority of PeopleTools user interface elements viewed by the end user are stored in the database, PeopleTools' executable modules do contain some translatable elements. When running PeopleTools components (such as the application server, PS/nVision and PeopleSoft Query) on Windows, these translations are read from alternate language DLLs. An alternate language DLL is a file that stores translated strings for an associated PeopleTools DLL or EXE file (module). At runtime, the system uses the resources in the alternate language DLLs that correspond to the user's language preference.

The PeopleSoft system uses a naming convention that includes the name of the base executable (the executable with which the alternate language DLL is associated) and the language code that identifies the language of the alternate language DLL. An alternate language DLL is named as follows:

PSZZZXXX.DLL

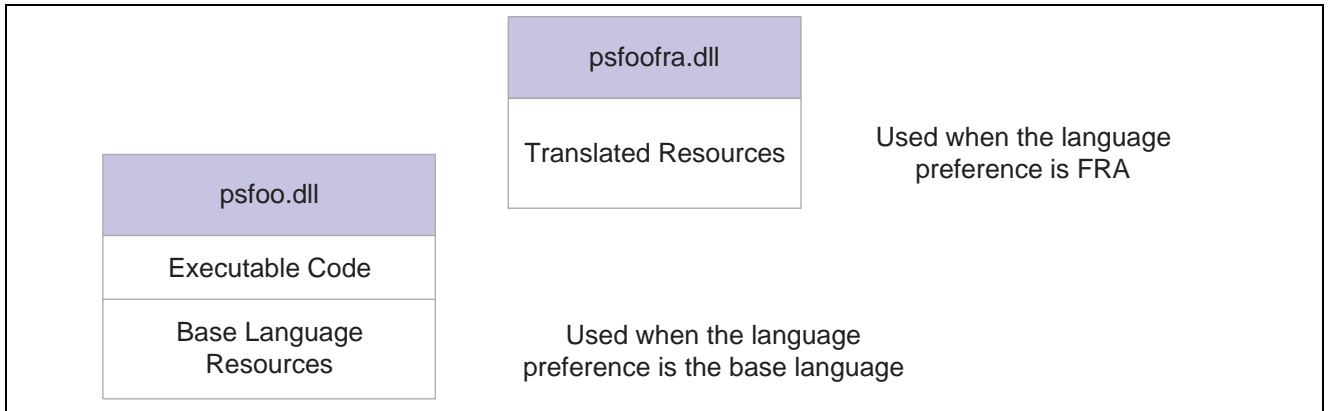
ZZZ

Three-character code that identifies the PeopleTools module associated with the alternate language DLL.

XXX

Three-character code that identifies the language of the alternate language DLL.

The alternate language DLL is used only to store translated resources. All code is executed from the base-language executable. The following diagram illustrates the structure of a base-language module (PSFOO.DLL) and its French alternate language DLL (PSFOOFRA.DLL):



Base language DLL and alternate language DLL

The system uses the resources that are stored in the alternate language DLL that matches the user’s language preference, if the DLL exists. Otherwise, it uses the resources that are stored in the base executable file.

Locating Resource Directories

Alternate language DLLs must be present on all application servers. Windows clients in a three-tier installation also must have the alternate language DLLs present.

Windows NT application servers and Windows client machines use Windows resources from the base language and alternate language DLLs. UNIX/Linux application servers, however, store resources for all languages and modules in a single file, PSAPPSERV.RES. This file contains the same resources as the alternate language DLLs.

The following table shows the content of the directories that are used to support translated resources. These directories are distributed with PeopleTools and contain user-customizable resource files and other supporting files that are needed to compile, bind, name, and copy alternate language resource DLLs.

Directories and Files	Purpose
\SRC\RES	The root directory for alternate language resource DLLs. The files in this directory include some batch files that are used in constructing alternate language DLLs.
\SRC\RES\ENG	A prototypical alternate language development directory for the English language. There is one of these directories for each alternate language. Copy this directory for each new alternate language that you create.

Directories and Files	Purpose
\SRC\RES\ENG\INC	Holds header (.H) files that are common to several DLLs and various icons and bitmaps.
\SRC\RES\ENG\XXXXX	<p>Within this directory, there is a directory for most PeopleTools DLL and EXE files found in \SRC\BIN. These directories contain the following file types:</p> <ul style="list-style-type: none"> • *.RCX: string table resource files. • *.RC: menu and dialog box resource files. • *.CUR: cursors. • *.H: header files containing resource identifiers. • *.BMP: bitmap files used for icons and other graphical components. • *.ICO: icon files for Windows Explorer.

See Also

[Chapter 14, “Translating PeopleTools,” Using MAKEUNIX.BAT, page 214](#)

Translating Resource Files

To translate resource files into a new language, we recommend that you start by copying the source code for the English translations provided by PeopleSoft and use this as a starting point for your new language. Alternatively, if you are only making minor modifications to an existing translation provided by PeopleSoft (such as to change terminology), you can use any of the languages provided by PeopleSoft as your starting point. This chapter assumes that you are using the English translations as your start, and that you’re adding a new translation.

Note. Keep in mind that PeopleSoft provides pre-translated PeopleTools alternate language DLLs for many languages, including French, Spanish, Dutch, Portuguese, Japanese, German, Italian, Danish, Polish, Swedish, Traditional Chinese, Simplified Chinese, and Canadian French. You must follow these steps only if you want to translate PeopleTools into a language that is not provided by PeopleSoft or if you want to modify one of the provided translations.

This section discusses how to:

- Translate English resource files.
- Use MAKEALTL.BAT.
- Use MAKEUNIX.BAT.

Translating Resource Files

To translate resource files:

1. Determine the three-letter designation of the new language.

In this example, *NRS* is the designation for Norwegian. For consistency in referring to languages throughout the system, use one of the standard code values stored in the Translate table for the LANGUAGE_CD field. The PeopleSoft system does not use standard ISO codes for languages, but three-letter codes instead. You can decide the appropriate three-letter code for your language, but you should ensure that it is consistently used across PeopleTools whenever you use this language.

See [Chapter 13, “Adding New Languages,” page 199](#).

2. Copy the contents of \SRC\RES\ENG to the target resource directory.

For Norwegian, the target resource directory is \SRC\RES\NRS.

3. Use Microsoft Developer Studio to convert the English resources to the alternate language.

A text editor is adequate to change string resources contained in the *.RCX files, however when modifying *.RC files that contain Windows dialog boxes and other graphical elements, you should use the resource editor provided in Microsoft Developer Studio bundled with Microsoft Visual C++. The resource editor enables you to size and position elements in dialog boxes, edit bitmaps, cursors and other graphic components of the resource files.

4. Set up the environment variables.

Before compiling your translated resource files, you must set the appropriate environment variables for the C++ resource compiler. Typically, Microsoft Visual C++ provides a batch file, VCVARS32.BAT, which sets the variables for you. In addition to the variables set in this batch file, set the environment variable TOOLBIN to the <PS_HOME>\BIN\CLIENT\WINX86 directory of your file server, and append the <PS_HOME>\SRC\RES\ENG\INC directory to the environment variable INCLUDE.

5. Compile alternate language DLLs using MAKEALTL.BAT.
6. (Microsoft Windows NT application servers only) Copy the resultant alternate language DLLs to the BIN\SERVER\WINX86 directory of your Windows NT application server.
7. (UNIX/Linux application servers only) Compile UNIX/Linux application server resources and transfer to the UNIX/Linux application server.

If you are using a UNIX/Linux application server, you must run MAKEUNIX.BAT to compile the resource files into a format that is readable by the UNIX/Linux application server.

Note. If you have changed any menu item text, update the Security settings to give users access to the menu items.

See Also

[Chapter 14, “Translating PeopleTools,” Using MAKEALTL.BAT, page 213](#)

Using MAKEALTL.BAT

The MAKEALTL.BAT file calls the Microsoft Visual C++ Resource Compiler and Linker to compile translated resources into a Windows DLL. Before running MAKEALTL.BAT, ensure that the environment variables required by the Resource Compiler and Linker are set correctly in your current DOS window. These environment variables are as follows:

Environment Variable	Description	Example
%path%	Must include your Microsoft Visual C++ executables directory.	C:\MSDEV\BIN
%include%	Must include your Microsoft Visual C++ INC directory and the Microsoft Foundation Classes (MFC) BIN directory. It must also include a copy of the PeopleTools resource include directory.	C:\MSDEV\INC C:\MSDEV\MFC\INC C:\PT810\SRC\RES\ENG\INC
%lib%	Must include your Microsoft Visual C++ library directory.	C:\MSDEV\LIB
%toolbin%	Must point to the directory containing your PeopleTools executables.	C:\PT810\BIN\CLIENT\WINX86

Once you have verified that the variables in the table above are correct, you can run MAKEALTL.BAT.

To run MAKEALTL.BAT:

1. In a DOS window, go to the SRC\RES directory.
2. Run the MAKEALTL batch file with <language> and <PeopleTools module> parameters.

This process compiles the resources into language-specific DLLs and copies the files to your PeopleTools executables directory.

The syntax of the command is as follows:

```
MAKEALTL <language> <PeopleTools module>
```

- <language> is the three-letter code for the language that you are compiling (such as NRS).
- <PeopleTools module> is the name of the directory that contains the resource files that you have translated (such as PSSYS).

Pass *ALL* in place of the directory name to compile all PeopleTools modules in the language that you specified.

Using MAKEUNIX.BAT

MAKEUNIX.BAT compiles the translated resources into a single file that is readable by the PeopleSoft UNIX/Linux application servers. It reads the same translated resources as MAKEALTL.BAT, but instead of producing a separate DLL for each language/module combination, MAKEUNIX.BAT produces a single indexed resource file called PSAPPSRV.RES.

To run MAKEUNIX.BAT, you must also have the Microsoft Visual C++ development environment installed on your workstation. MAKEUNIX.BAT calls the C++ compiler to parse the resource files from all languages and to create PSAPPSRV.RES. In addition to the environment variables that are required to run MAKEALTL.BAT, MAKEUNIX.BAT requires that you set the %PSVER% environment variable. Set this variable to the base directory in which you installed PeopleTools.

If you have added any new languages as part of your PeopleTools translations, you must edit MAKEUNIX.BAT to include the new languages so that they are included in the PSAPPSRV.RES file.

To run MAKEUNIX.BAT:

1. Open a DOS command prompt window.
2. Change the directory to your \PT810\RES directory.
3. Run the MAKEUNIX batch file.

MAKEUNIX doesn't require that you pass the module or language name arguments; it compiles the UNIX/Linux application server resources for all languages and modules in one pass.

The output from MAKEUNIX.BAT, and PSAPPSRV.RES is placed in <PS_HOME>\APPSERV\UNIX. You must transfer PSAPPSRV.RES to the BIN directory of your UNIX/Linux application server using a network file transfer utility, such as FTP.

Translating Objects

Not all translatable elements used internally by PeopleTools can be contained in resource files. Some strings referenced by Java and other components of PeopleTools are stored in Java-style property files. The PeopleSoft Pure Internet Architecture makes use of images for many navigational and user interface elements. Some of these images contain text which must also be translated.

This section discusses:

- Translating property and text files.
- Translating images.

Translating Properties Files

Java components of PeopleTools such as the servlets on the Web Server derive some strings from Java-style properties files. Java properties files are simple text files that are read at runtime by the servlet and contain strings that are displayed to the user, typically in error messages.

These files can be translated using any standard text editor, and are located on the Web Server. The following files contain translatable text:

Base Properties File	Translated Properties File
text.properties	text_<language_code>.properties
errors.properties	errors_<language_code>.properties

The base version of the file (without a language code) contains the English version of the strings. Copy this file to a new file containing the language code of your target language before translating the contents. For example, if you are translating the resources into Norwegian, you would create text_NRS.properties and errors_NRS.properties, and update these files with Norwegian translations of the contents.

See [Chapter 13, "Adding New Languages," page 199](#).

When translating into a language that cannot be represented by the Latin-1 character set, such as Japanese, Chinese and Korean, the contents of the Java properties files must be converted into escaped Unicode character references. The Sun Java Development Kit (JDK) contains the native2ascii utility that you can use to convert Unicode character references in a Java properties file into a native character set and vice versa. Refer to the native2ascii utility documentation in the JDK for more information.

Translating Images

The PeopleTools Internet Architecture uses many images as part of its user interface component, and some of these images contain text. You will typically translate only images that contain textual elements or elements that are specific to one language or culture.

Translating an Image

To translate an image:

1. Change the PeopleSoft Configuration Manager language preference to the target language.
2. Open PeopleSoft Application Designer.
3. Open the image.

Choose File, Open and select the image you want to open.

4. Export the image to a file.

From the File menu select Export Image, and select a filename and image format to save the image to disk.

5. Use a graphics editor to modify the saved image file and update the translation.
6. From the Edit menu select Update Image.

A standard Open dialog box appears.

7. Open the file with the translated image.

The new graphic replaces the original graphic in the image.

8. Save the image.

The translated image is saved into the related language table under the language that you selected in the Configuration Manager in the first step.

CHAPTER 15

Translating Application Definitions

This chapter provides an overview of PeopleSoft application definition translation processes and discusses how to:

- Use the Translation Designer feature.
- Use the translation pages.

Understanding Application Definition Translation

If you are running PeopleSoft applications with a multiple language user interface, you should translate any customizations you make to the product using PeopleTools into each of the languages that you are using. PeopleTools provides a full suite of tools to help you translate the application user interface; these are the same tools that PeopleSoft uses internally to provide the translations we ship on the Global Multi Language CDs.

The main definitions requiring translation in order to provide a fully multilingual user interface are pages and fields. However, PeopleTools allows the translation of many other definitions; everything that the end user sees via the PeopleSoft Pure Internet Architecture can be translated into multiple languages.

You can translate PeopleTools definitions using PeopleSoft Application Designer, PeopleSoft Tree Manager, or the translations utilities. The translations utilities are a particularly powerful option for translating fields, translate values, and hard-coded text on pages (that is, text that is not derived from the field description). Given the definitional approach to PeopleTools applications, however, the order in which you translate objects is critical in order to obtain the most leverage from your translation work.

Using the Translation Designer Feature

The Translation Designer feature within PeopleSoft Application Designer provides an efficient mechanism for translating PeopleSoft pages, which are the highest volume and highest profile definitions.

Page definitions include:

- Field labels.
- Page text (not derived from field labels).
- Translate values (which normally appear either as radio button labels or as values in dropdown lists).

Translation Designer facilitates translations with an easy-to-use grid where you can view the base-language text and enter a translation simultaneously, enabling translators to review their translations side-by-side instead of simply overwriting source text with a translation.

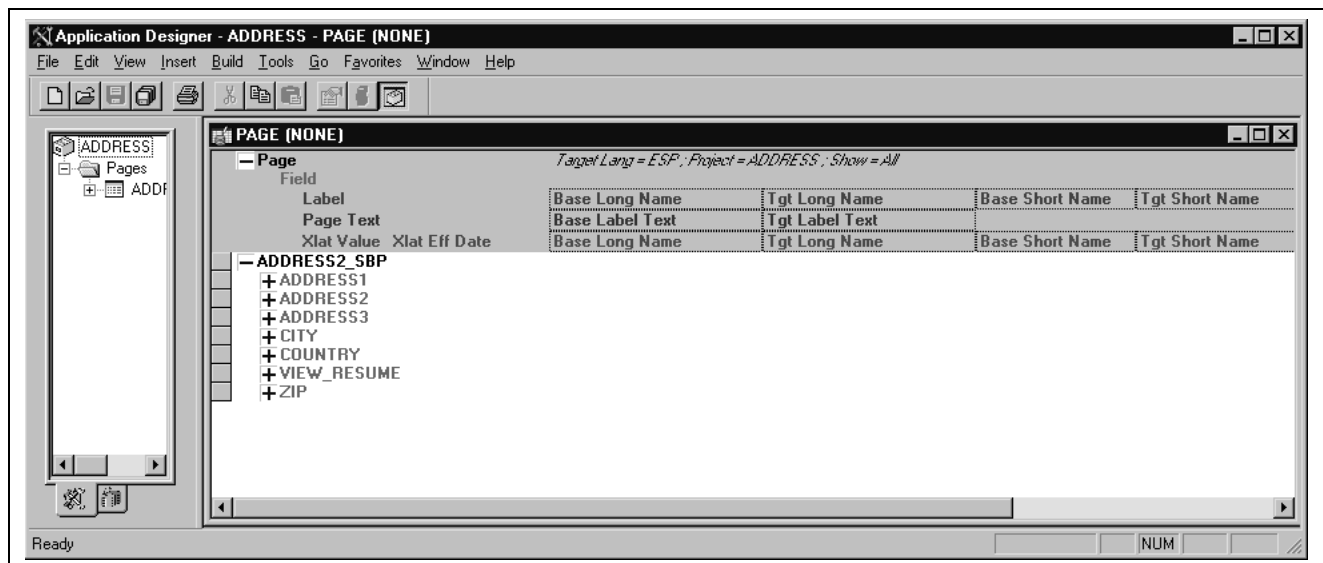
This section discusses how to:

- Open Translation Designer.
- Use Translation Designer display options.
- Work in Translation Designer.
- Translate system definitions.

Opening Translation Designer

Translation Designer shows all the translatable definitions in your current PeopleSoft Application Designer project, along with their base-language and translated text.

The grid contents are based on the PeopleSoft Application Designer project that is open when Translation Designer is started. This association with PeopleSoft Application Designer projects is a handy mechanism for organizing your translation effort. However, it also means that, to use Translation Designer, you must have a project open, and it must contain one or more pages. If you modify the project, Translation Designer won't pick up the new objects in the project until you save the changes and refresh.



Translating Page Definitions in Translation Designer

To open Translation Designer:

1. Set your language preference to the target language.
Because you use the PeopleSoft Application Designer to access Translation Designer, set your language preference using PeopleSoft Configuration Manager.
2. Open PeopleSoft Application Designer.
3. Create or open the project containing the definitions to be translated.

It makes sense to use projects with the non-base language versions of any pages. That way you're sure to include all definitions on the pages. If you are creating a new project, you must save it before opening Translation Designer. It is a good idea to keep the number of pages in the project being translated to a minimum, as all the fields and other translatable definitions on each page in the project are loaded into memory when Translation Designer starts. A project with a lot of pages may take some time to open due to the sheer volume of fields on the page. You should try to keep the number of pages in a project to be translated using Translation Designer under 20.

4. Open Translation Designer.

- To translate the pages in your project, from the Tools menu select Translate and choose Translate Page Definitions.

This is the most versatile option because it shows you all translatable definitions that are associated with a page. You can use it to translate field labels, panel text (not derived from field labels), and translate values—all in one grid.

- To translate only the fields in your project, from the Tools menu select Translate and choose Translate Fields.

This option lets you translate fields that are directly included as definitions in your project. Fields that are part of records or pages in your project are not included unless they are also explicitly included in your project as fields.

- To translate only the translate values in your project, from the Tools menu select Translate and choose Translate Xlats.

You can also access all of these commands in the popup menu that appears when you right-click in the project workspace.

After you select one of the translate options, Translation Designer appears in the object workspace.

Depending on which translation grid you opened, different translatable definitions from the active project appear in the grid. All definitions are arranged hierarchically: pages (if included) are at the top level; fields are at the next level; and field labels, panel text, and translate (xlat) values are at the lowest level.

Use standard tree controls to expand and collapse the view in order to display definitions at different levels. You can also collapse or expand all levels of the tree by right-clicking on the tree node and selecting the appropriate option from the popup menu.

5. (Optional) Arrange the window to maximize Translation Designer.

If you want more space for the translation grid, maximize the PeopleSoft Application Designer window and then maximize Translation Designer within the window. For even more space, hide the other frames in the PeopleSoft Application Designer window. To toggle the display of the project workspace, from the View menu select Project Workspace, or press ALT+0 (zero). To toggle the display of the output window, from the View menu select Output Window, or press ALT+1.

See Also

[Chapter 2, “Controlling International Preferences,” Setting Up Language Preferences for Microsoft Windows-Based PeopleTools, page 17](#)

Using Translation Designer Display Options

This section discusses how to:

- Use the information bar.
- Expand and collapse nodes.
- Choose which rows to view.
- Resize columns and rows.

Using the Information Bar

Across the top of the translation grid, an information bar displays helpful information about the project. The target language, current project, and current viewing option (all, translated, untranslated, or modified) are set in italic.

On the left side of the bar, you can see the hierarchical organization that is used to display definitions in the grid: pages (at the highest level), fields (at the next level), and translatable definitions—field labels, page text, and translate values—at the lowest level.

Each definition is color-coded to remind you with which definition type you’re working.

- Fields: blue.
- Field labels: light purple.
- Page text (not derived from a field label): dark purple.
- Translate values: green.

Further, the following graphical cues help you understand the display:

- All fields on the page are displayed, even if they are hidden fields.
- The default label ID appears in italics, but not boldface.
- Any label ID that is used but that is not the default label appears in boldface italics.
- Page text from hyperlinks is displayed.
- If the same field has multiple label IDs on the same page, it appears only once for that page.

Field	Label	Base Long Name	Tgt Long Name	Base Short Name	Tgt Short Name
Page	Page Text	Base Label Text	Tgt Label Text	Base Short Name	Tgt Short Name
	Xlat Value	Xlat Eff Date			
ADDRESS2_SBP	Page - Group Box	Address	Address		
ADDRESS1	<i>Lbl - ADDRESS1</i>	Address Line 1	Dirección1	Address 1	Dirección1
ADDRESS2	<i>Lbl - ADDRESS2</i>	Address Line 2	Dirección2	Address 2	Dirección2
ADDRESS3	<i>Lbl - ADDRESS3</i>				
ADDRESS_TYPE	<i>Lbl - ADD_TYPE</i>	Address Type	Address Type	Type	Type
	Xlt - H 2000-01-01	Home	Home	Home	Home
	Xlt - M 2000-01-01	Post Office Box	Post Office Box	PO Box	PO Box
	Xlt - O 2000-01-01	Other	Other	Other	Other
	Xlt - W 2000-01-01	Work	Work	Work	Work
CITY	<i>Lbl - CITY</i>	City	Cuidad	City	Cuidad
COUNTRY	<i>Lbl - COUNTRY</i>	Country	País	Countryt	País
ZIP	<i>Lbl - ZIP</i>	Postal Code	Cd.Postal	Zip	Cd.Postal
	<i>Lbl - ZIP2</i>	ZIP Code		Zip	

Color-coded definitions

In this example, notice that there is a group box with page text that is not derived from a field label. The ADDRESS_TYPE field has a set of associated translate values. All other translatable definitions are field labels.

Notice also that the default label ID is italicized but not bold, while any non-default label IDs used are in bold italics. This is apparent when you look at the ZIP field, which has two labels. The other field labels appearing in the grid are associated with the field, but not used on the page being translated; they are displayed primarily so the translator can see the context in which they are translating the label and ensure consistency of terminology across labels.

Expanding and Collapsing Nodes

The left side of the translation grid is a hierarchical tree control that displays all the definitions on the page.

- +** Click once to expand the definition by one level.
- Click once to collapse the definition by one level.

To expand the entire tree structure, from the View menu select Expand All. To collapse the entire tree structure, from the View menu select Collapse All. Right-clicking anywhere in the grid displays a popup menu that contains these options. Another option on the popup menu, Expand Current Definition, expands the selected definition by one level.

Choosing Which Rows to View

Right-click anywhere in the translation grid to see a popup menu that lets you choose which rows are displayed in the grid:

- All** Shows all translatable definitions.
- Translated** Shows only the definitions that have already been translated—that is, where the base language label and target language label do not match.
- Untranslated** Shows only the definitions that have not been translated—that is, where the target language label is blank or the base language label and target language label match.
- Modified** Shows the definitions that have been modified in the translation grid.
Translation Designer defaults to this mode when it encounters a save conflict.

Keep in mind that the options you select in the popup menu work together. If you select Expand Current Object while Untranslated is activated, you might not see anything. This means that everything in the current definition is translated.

Resizing Columns and Rows

To change the column width, place the cursor over the column divider on the information bar. When the cursor changes, drag the column divider to the desired position. You can resize only translatable columns; you cannot resize the columns that display the hierarchical tree of page definitions. Use the same technique to change row height.

Working in Translation Designer

This section discusses how to:

- Enter translations.
- Navigate in Translation Designer.
- Search and replace.
- Use PeopleSoft Application Designer functionality.
- Integrate with other applications.

Entering Translations

The working area of Translation Designer consists of four columns of data: Base Long Name, Tgt Long Name (target long name), Base Short Name, and Tgt Short Name (target short name).

As the column names imply, the two base language columns display the base language labels for the definitions in the grid. To translate the labels, type the translation into the Tgt Long Name and Tgt Short Name fields.

The grid incorporates standard text editing functionality. Cut, copy, and paste operations are available under the Edit menu. To force a line break within a label (so that the text wraps on the page), press CTRL+ENTER.

The first time that you translate a particular field label, the same translation is entered into any other occurrence of that field in the current grid. This ensures consistency and saves time.

The grid automatically limits the length of the text you enter based on the maximum length of the fields. When you reach the maximum length, you cannot add any more characters.

As with any PeopleSoft Application Designer definition, changes are not permanent until you save them.

Note. To save related language data, you must have values in all fields. You cannot partially translate a single PeopleTools definition. For example, if you translate the long name but not the short name for a field, the short name defaults to the base language short name, and this value is saved in the related language table. Similarly, if you translate some but not all translate values for a particular field, any untranslated values pick up the base language text, which is then saved to the related language table.

Navigating Translation Designer

Click to move to a cell or node in Translation Designer, or use the standard Windows keyboard shortcuts:

- Press TAB or ENTER to move to the right; press SHIFT+TAB to move to the left.
- Use any of the arrow keys to move from cell to cell.

For example, the UP ARROW key moves you to the cell above the current cell.

- To move from the text cells to the tree, press the LEFT ARROW key.
- To move from the hierarchical tree to the text cells, press TAB.

You can use the RIGHT ARROW key only when there are text cells in the same row—that is, if the current node is a translatable definition.

- Use the UP ARROW and DOWN ARROW keys to move around in the hierarchical tree—for example, from a field to an adjacent field or to a page.

To move to a lower level, first press TAB to move into the main Translation Designer. Then move up or down to the desired node. Then use the LEFT ARROW key to move back into the tree.

- If there are several pages in your project, scroll directly to the node for a particular page by right-clicking that page in the project workspace and from the menu select Translate and choose Translate Page Definitions from the popup menu.

Searching and Replacing

To search for a specific word within a single column:

- Place the cursor anywhere in that column, and select Edit, Find and Replace In Current Column.

If you are searching a non-editable column, from the Edit menu choose Find in Current Column.

- The Find and Replace dialog boxes provide standard search and replace functionality, including the Find Next, Replace, and Replace All buttons. In either dialog box, select the Match case check box if you want to enable case-sensitive searching.
- The Find option provides enables you to search up or down the column.
When you reach the top or the bottom of the column, the search does not cycle back through the column. Therefore, if you want to search the entire column, ensure that you're in the top row when you press Find Next.
- The Replace function always searches down the column.
If you want to replace all occurrences in the column, ensure that you're in the top row when you press Replace.

Using PeopleSoft Application Designer Functionality

Because Translation Designer is integrated into the PeopleSoft Application Designer, you can use all standard PeopleSoft Application Designer features while Translation Designer is open. Some features that are particularly useful during translations are:

- Using the Find Definition References option to research where in the system a particular definition is used. To use this option, from the Edit menu select Find Definition References.
- Opening a translated page in order to realign translated definitions.
- Opening definitions to see the properties.

Remember, as long as you're logged on using a non-base language, modifying labels in the definition updates the related language tables, not the base language tables.

There are several ways to open a definition:

- Use standard PeopleSoft Application Designer functionality to open the definition from the project window.
- From the File menu, select Open.
- Double-click or right-click the definition within Translation Designer and select View Definition from the popup menu that appears.

When you have selected page text that is not associated with a field definition, the page itself opens.

If there is a version of the page for the user's language preference, that version appears; if not, the base language page appears.

Note. Security controls access to different PeopleSoft Application Designer definitions. If you cannot open a particular type of definition from within Translation Designer, you may not have the necessary level of access.

See *PeopleTools 8.45 PeopleBook: PeopleTools Security*

Integrating with Other Applications

Because Translation Designer emulates a Microsoft Excel spreadsheet, you can use the standard Windows copy and paste tools to copy translations from Translation Designer into Excel or another Windows application. This is useful when running spell check against your translations or when saving your translations in a spreadsheet for offline review.

To copy the entire contents of the currently open Translation Designer into the clipboard:

1. Select the entire grid.
From the Edit menu choose Select All or click in the top left cell of the grid border.
2. From the Edit menu select Copy.

Note. With Translation Designer, you can copy an entire grid to the clipboard, but you can paste only individual field values back into the grid. You cannot paste the entire grid into Translation Designer after reviewing it in another application.

Using the Translation Pages

This section provides an overview of the translation pages and discusses how to:

- Translate page definitions.
- Translate messages.
- Translate menus.
- Translate business processes.
- Translate PeopleSoft Process Scheduler.
- Translate mobile data.
- Translate portals.
- Translate trees.
- Translate records.
- Translate queries.
- Translate strings.
- Translate PeopleSoft Application Engine descriptions.
- Translate review page text.
- Translate application data.
- Translate HTML definitions.

Understanding the Translation Pages

For definitions that cannot be translated via Translation Designer, PeopleTools contains several translation pages that are especially useful if the translator has limited experience using PeopleTools. A translator who has good technical skills and who is familiar with PeopleTools may prefer to translate using the design tools as described later in this chapter.

To locate the translation pages, select PeopleTools, Translations, Translate System Definitions. The translation pages will help you to translate system definitions, including records, menus, menu items, components, component interfaces, queries, message channels, messages, message definitions, portal definitions, process definitions, PeopleSoft Application Engine programs, Business Interlinks, review page text, and HTML definitions.

You should translate PeopleTools definitions in the following order to ensure the least amount of duplication during your translation process:

1. Translate pages, messages, and menus—in any order (using Translation Designer).
2. Translate business processes, Process Scheduler, mobile pages, and portals—in any order.
3. Translate trees, records, queries, strings, application data, and PeopleSoft Application Engine—in any order.

Translating Page Definitions

All page objects except the HTML definitions can be translated directly in Translation Designer. However, you can also translate pages, translate values, and fields directly in PeopleSoft Application Designer if you are familiar with the designer tools.

Definition	Base Table	Related Language Table	Where to Translate
Field Labels	PSDBFLDLABL	PSDBFLDLABLLANG	Translation Designer
Translate Values	PSXLATITEM	PSXLATITEMLANG	Translation Designer
Page Definitions	PSPNLDEFN	PSPNLDEFNLANG	Translation Designer
Page Fields (Page text)	PSPNLFIELD	PSPNLFIELDLANG	Translation Designer
Page Buttons	PSPNLBTNDATA	PSPNLBTNLANG	Translation Designer
Page HTML	PSPNLHTMLAREA	PSPNLHTMLLANG	Application Designer

Note. Don't be confused by the term *translate value* or *translate table*. The translate table is a common table that is used throughout PeopleTools and PeopleSoft applications and is not used for translation purposes.

Translating Translate Values

To translate values from the Translate table in PeopleSoft Application Designer:

1. Change the PeopleSoft Configuration Manager language preference to the target language.
2. Open PeopleSoft Application Designer.
3. Open the field whose translate values you want to access.
4. From the File menu, select Definition Properties.
5. Access the translate values tab in the Field Properties dialog box.
 - a. Click the Properties icon (or right-click in the field definition and choose Field Properties).
 - b. Click the Translate Values tab (this tab is visible only for fields with a length of four or less).
If the descriptions have not been translated, they appear in the base language.
6. Translate the descriptions into the target language.
 - a. For each translate value, click the Change button to display the Change Translate Table dialog box.
 - b. Translate the Long Name and Short Name fields.
Don't change any other fields.
 - c. Click OK to accept the changes and close the dialog box.
7. Click OK to close the Field Properties dialog box.
8. Save the field.

Translating Messages

When you translate messages, you translate the message set description, the messages, and the detailed explanations of the messages into a specific target language.

The following table lists the objects, base tables and related language tables related to translating message sets and message catalogs.

Object	Base Table	Related Language Table	Where to Translate
Message Sets	PSMSGSETDEFN	PSMSGSETLANG	Translate Messages page.
Message Catalog	PSMSGCATDEFN	PSMSGCATLANG	Translate Messages page.

To translate messages, use the Translate Message page. To access this page, select PeopleTools, Translate System Definitions, Messages.

Translate Messages

Selection Criteria

*Message Set Number: Message Number: To: *Language Code:

Messages Find | View All First 1 of 1 Last

Base/Target Last Updated:

Base Message Text:

Base Explanation:

Target Message Text:

Target Explanation:

Translate Messages page

To translate messages:

1. Select PeopleTools, Translations, Translate System Definitions, Messages. The Translate Message page displays.
2. Select a message set with which to work.
Enter a Message Set Number in the field or click the Prompt button to search for a specific message set number.
3. Select messages within the message set with which to work.
 - a. To select all messages within the message set, leave the Message Number field blank.
 - b. To select a range of messages within the message set, enter the range in the Message Number fields.

4. Select the target language.
 - a. From the Language Code dropdown list, select a target language.
 - b. Click the Get Messages button.

The message set description you selected appears under the Message Set Number.

The first message in the set or in the range you selected displays in the Messages section. The Base Message Text and Base Explanation fields display the message and its explanation in the base language.

Use the arrow buttons or the First and Last links on the Messages bar to navigate to the messages retrieved.
5. Translate the message set description into the target language.
 - a. Click the Information icon next to the Get Messages button. The Add Message Set page displays.
 - b. Enter a Description in the target language for the message set.
 - c. Enter a Short Description for the message set in the target language. The short description must be 10 characters or less.
 - d. Click OK to return to the Translate Messages page.
6. Translate the message text and explanation into the target language.
 - a. In the Target Message Text field, enter a translation in the target language for the information in the Base Language Text field.
 - b. In the Target Explanation field, enter a translation in the target language for the information that appears in the Base Explanation field.
7. Click Save.
8. Repeat steps 6 and 7 for each message in the message set.

On the Messages bar, use the Forward or Back arrows or the First and Last links to navigate to other messages within the set to translate.

Translating Menus

The following table lists where to translate menu and component objects:

Object	Base Table	Related Language Table	Where to Translate
Menu Definitions	PSMENUDEFN	PSMENUDEFNLANG	Translate Menu page.
Menu Items	PSMENUITEM	PSMENUITEMLANG	Translate Menu Item page.
Component Descriptions	PSPNLGRPDEFN	PSPNLGDEFNLANG	Translate Components page.
Components	PSPNLGROUP	PSPNLGROUPLANG	Translate Components page.

This section discusses how to:

- Translate menu definitions.
- Translate menu items.
- Translate folder tabs in PeopleSoft Application Designer.
- Translate folder tab labels using the translation utilities.

Translating Menu Definitions

Use the Translate Menus page to translate menu definitions and menu group labels.

In a portal, the names that appear in the menu are actually content references. You must translate these using the Translate Menus Page. Use PeopleSoft Application Designer to change only popup menus.

To translate menus:

1. Select PeopleTools, Translations, Translate System Definitions, Menus.

The Translate Menu page displays. The Translate Menus page enables you to select a set of menu definitions and translate the labels for its menu and menu group into a specific target language.

2. In the Target Language field, select the language into which you want to translate the message set.
3. In the Menu Name field, select a set of menu definitions.

Leave the Menu Name field blank if you want to select all the menus in the system.

If you want to work through the menus alphabetically, enter the first characters of the menu name in the Menu Name field. If you want to translate a specific menu, enter the menu name in the field or click the Search button to prompt for a specific menu.

4. Click the Get Menu button to bring the menu descriptions into the menu list.
5. Translate the menu label and menu group into the target language.

The base-language descriptions for each label appear on the left; enter the translations in the fields on the right.

If you want to include accelerator keys for use with Windows client navigation, position the accelerator key ampersand to the left of the character that will serve as the accelerator key for the menu or menu group.

6. Save the page.

Translate Menu Items

To translate menu items:

1. Select PeopleTools, Translations, Translate System Definitions, Menu Items.

The Translate Menu Items page displays. The Translate Menu Items page enables you to select a set of menu definitions and then translate the labels for its menu bars and menu items into a specific target language.

2. In the Target Language field, select the language into which to translate the menu bar and menu item labels.
3. In the Menu Name field, select a set of menu definitions.

Leave the Menu Name field blank if you want to select all the menus in the system.

If you want to work through the menus alphabetically, enter the first characters of the menu name in the Menu Name field. If you want to translate a specific menu, enter the menu name in the field or click the Search button to prompt for a specific record.

4. Click the Get Menu Items button to bring the menus into the Menu Item List.

5. Translate the bar label and item label into the target language.

The base-language descriptions for each label appear on the left; enter the translations in the fields on the right.

The translated menu bar label appears at runtime if the user's language preference is set to the target language. It will also appear if one or more menu items that appear in the menu bar have a menu bar label translated into the target language.

To include accelerator keys for use with Windows client navigation, position the accelerator key ampersand to the left of the character that will serve as the accelerator key for the menu or menu group.

After you translate a menu bar label and exit the field, you must select whether to use the same translation for all occurrences of the bar label:

- Click *Yes* if you want the translation you just entered to be copied to each of the menu items in the list that share the same bar label. This is merely a data entry shortcut; no translations are saved until you explicitly save the page.
- Click *No* if you want to leave some of the menu items untranslated.

6. Save the page.

Translating Folder Tabs in PeopleSoft Application Designer

Folder tabs should be translated directly in the Translate Components page. However, if you are familiar with design tools you can also translate folder tabs directly in the PeopleSoft Application Designer, Translation Designer.

Component definitions include two labels for each page in the component: an item label and a folder tab label. In the PeopleSoft Pure Internet Architecture, only one of these is visible to the user. If a folder tab label exists, it is used on the folder tab for the corresponding page; the item label is ignored. If no folder tab label exists, the item label is used as the folder tab label.

To translate folder tab labels in PeopleSoft Application Designer:

1. Change the PeopleSoft Configuration Manager language preference setting to the target language.
2. Log in to the PeopleSoft system.
3. Open PeopleSoft Application Designer.
4. Open the component definition.
5. Translate the item labels and folder tab labels into the target language.

Edit the labels directly in the Item Label and Folder Tab Label columns on the grid.

If the cell in the folder tab label column is blank, the item label appears in the folder tab.

If there is both an item label and a folder tab label, be aware that only the folder tab label is visible in the PeopleSoft Pure Internet Architecture. The item label appears only on Windows menus.

For items that users access in Windows menus, you may want to include ampersands within the item label text to create accelerator keys.

6. Save the component definition.

Translating Folder Tab Labels Using Translation Utilities

To translate folder tab labels using the translation utilities:

1. Select PeopleTools, Translations, Translate System Definitions, Components.

The Translate Components page displays. The Translate Components page enables you to select a set of component definitions and then translate the labels for the pages that make up the component.

There are two translatable labels for each page in the component: an item label and a folder tab label.

In the portal, only one of these labels appears to the user. If a folder tab label exists, it is used on the folder tab for the corresponding page. The item label is ignored. If no folder tab label exists, the item label is used as the folder tab label.

In Windows, item labels become part of the navigational structure, so if the item label and folder tab label are different, they should both be translated. However, because folder tabs use the item label when the folder tab label is missing, it is common to have item labels only.

2. In the Target Language field, select the language into which to translate the component labels.
3. In the Component Name edit box, select a set of component definitions.

Leave the Component Name field blank to select all the components in the system.

If you want to work through the components alphabetically, enter the first characters of the component name in the Component Name field. If you want to translate a specific component, enter the component name into the edit box or click the Search button to prompt for a specific record.

4. Click the Get Component button to bring the components into the component list.

If a component item has no folder tab text, the item label appears in the folder tab. In these cases, you can leave the translation for the folder tab text blank.

5. Translate the item label and folder tab into the target language.

The base-language descriptions for each label appear on the left; enter the translations in the fields on the right.

If you want to include accelerator keys for use with Windows client navigation, position the accelerator key ampersand to the left of the character that will serve as the accelerator key for the menu or menu group.

6. Save the page.

Translating Business Processes

Translate menus before business processes, PeopleSoft Process Scheduler, and portals because the terms are related.

The following table lists where to translate business process objects:

Object	Base Table	Related Language Table	Where to Translate
Business Process Definition	PSBUSPROCDEFN	PSBUSPROCLANG	PeopleSoft Application Designer.
Activities	PSACTIVITYDEFN	PSACTIVITYLANG	PeopleSoft Application Designer.
Business Process Items	PSBUSPROCITEM	PSBUSPROCITEMLANG	PeopleSoft Application Designer.

Object	Base Table	Related Language Table	Where to Translate
Component Interfaces	PSBCDEFN	PSBCDEFNLANG	Translate Component Interfaces page.
Message Channels	PSCHNLDEFN	PSCHNLDEFNLANG	Translate Message Channels page.
Business Interlinks	PSIODEFN	PSIODEFNLANG	Translate Business Interlinks page.

This section discusses how to translate:

- Business process definitions (and their related objects).
- Component Interface messages.
- Message channels.
- Business Interlinks.

Translating Business Process Definitions (and their related objects)

To translate business process definitions:

1. Set the language in PeopleSoft Configuration Manager.
2. Open PeopleSoft Application Designer.
3. Open the Business Process Map.
4. Overwrite the English with the translation.
5. Save.

Translating Component Interface Messages

To translate component interface messages:

1. Select PeopleTools, Translations, Translate System Definitions, Component Interfaces.

The Translate Component Interfaces page displays. The Translate Component Interfaces page enables you to select a component interface and then translate the description of the component interface into a specific target language.

2. In the Target Language field, select the language into which to translate the message set.
3. In the Name field, enter a component interface.

Click the Search button to prompt for a list of component interfaces.

4. Click the Get Objects button to bring the component interface into the definition list.

A set of component interface names appears in the Object List.

5. Translate the component interface description into the target language.
 - a. Click the Update button to display the Add Message Set page.
 - b. Enter descriptions for the message set in the target language.

- c. Click OK to accept the change and return to the Translate Component Interfaces page.
6. Save the page.

Translating Message Channels

To translate channels:

1. Select PeopleTools, Translations, Translate System Definitions, Message Channels.
The Translate Message Channels page displays. The Message Channels page enables you to select a channel and then translate the descriptions of the channel into a specific target language.
2. In the Target Language field, select the language into which to translate the message set.
3. In the Channel field, enter a channel name.
Click the Search button to prompt for a list of channel names.
4. Click the Get Objects button to bring the message channels into the definition list.
A set of channel names appears in the Look Up List.
5. Translate the channel description into the target language.
Enter descriptions for the message set in the target language.
6. Save the page.

Translating Business Interlinks

To translate interface objects:

1. Select PeopleTools, Translations, Translate System Definitions, Business Interlinks.
The Translate Business Interlinks page displays. The Translate Business Interlinks page enables you to select the base description of an interface object name and then translate the description into a specific target language.
2. In the Target Language field, select the language into which to translate.
3. In the IO Name field, enter an Interface Object name.
Click the Search button to prompt for a list of interface object names.
4. Click the Get Objects button to bring the interface object names into the definition list.
A set of interface object names appears in the Look Up List.
5. Translate the base descriptions into the target language.
Enter descriptions for the interface object name in the target language.
6. Save the page.

Translating PeopleSoft Process Scheduler

PeopleSoft Process Scheduler has three language-sensitive definitions:

- Process type definitions.
- Job definitions.
- Process definitions.

The following table lists where to translate PeopleSoft Process Scheduler objects:

Object	Base Table	Related Language Table	Where to Translate
Process Definitions	PRCSDEFN	PRCSDEFNLANG	Translate Process Definitions page.
Process Types	PRCSTYPEDEFN	PRCSTYPEDEFNLNG	PeopleTools, Process Scheduler, Process Types.
Process Jobs	PRCSJOBDEFN	PRCSJOBDEFNLANG	PeopleTools, Process Scheduler, Jobs.
Process Recurrences	PRCSRECUR	PRCSRECURLANG	PeopleTools, Process Scheduler, Recurrences.

To translate a PeopleSoft Process Scheduler Types, Jobs and Recurrences definitions:

1. Sign on to the PeopleSoft system using the target language.
2. Select PeopleTools, Process Scheduler Manager to open PeopleSoft Process Scheduler.
3. Open the definition to translate.
 - To open a process type definition, select Process Type.
 - To open a job definition, select Job.
 - To open a process definition, select Recurrences.
4. Translate the descriptive text fields into the target language.
 - Translate the Description field.
5. Save the page.

Translating Process Definitions

To translate process definitions:

1. Select PeopleTools, Translations, Translate System Definitions, Process Definitions.
 - The Translate Process Definitions page displays. The Process Definitions page enables you to select a process name and type and then translate the long descriptions of the Process Definitions into a specific target language.
2. In the Target Language field, select the language into which to translate the Process Definitions.
3. Enter Process Name and process type into the appropriate fields.
 - Click the Search button to prompt for a list of names.
4. Click the Get Objects button to bring the Process Definitions into the definition list.
 - A set of Process Definitions appears in the Look Up List.
5. Translate the process definitions description into the target language.
 - Enter descriptions for the process definitions in the target language.
6. Save the page.

Translating Mobile Data

This section describes how to translate:

- Mobile page names.
- Mobile view labels.
- Mobile page items, such as fields.

Note. To translate mobile data, you must also translate the `\bin\mobile\res\<lang>\mptext.properties` file.

The following table lists where to translate mobile data.

Object	Base Table	Related Language Table	Where To Translate
Mobile page	PSMPDEFN	PSMPDEFN	Translate Mobile Page page.
Mobile page	PSMPDEFNDEV	PSMPDEFNDEVLANG	Translate Mobile Page page.
Mobile Page Items	PSMPCONTDEV	PSMPCONTDEVLANG	Translate Mobile Page Items page.

See [Chapter 13, “Adding New Languages,” Adding Text and Error Properties Files, page 206](#).

Translating Mobile Page Names and View Labels

Use the Translate Mobile Page Names page to translate Mobile page names and view labels.

Translate Mobile Page

Selection Criteria

Mobile Page Name:
Target Language:

Mobile Page List
Find | View All First ◀ 1 of 1 ▶ Last

Mobile Page Name: CURRENCY_CD_MP

Description: Currency Code CI

Long Description:

Translate Mobile Page

The fields with which you can work on the Translate Mobile Page are:

Mobile Page Name Name of the mobile page to translate.

Target Language	From the dropdown list select the target language for the translations.
Get Mobile Page	Click the button to refresh the page and display the Mobile Page List section, including the Description and Long Description fields.
Description	Enter the description for the page in the target language.
Long Description	Enter the long description for the page in the target language.
Mobile Page View Labels	Click the button to refresh the page and display the View Labels section, including the Detail View Label field and the List View Label field.
Detail View Label	Enter the detail view label in the target language.
List View Label	Enter the list view label in the target language.

To translate Mobile pages:

1. Select PeopleTools, Translations, Translate System Definitions, Mobile Pages. The Translate Mobile Page displays.
2. In the Mobile Page Name field, enter the name of the page to translate, or use the Lookup button to search for a page.
3. From the Target Language dropdown list, select the target language.
4. Click the Get Mobile Page button. The page refreshes and the Mobile Page List displays.
5. Translate the Mobile page name:

Note. If available, the descriptions that you can translate display next to the fields in the base language.

- a. In the Description field, enter a translation for the page description in the target language.
- b. In the Long Description field, enter a translation for the long description in the target language.

Note. A default description in the target language may display in the field. If so, clear the field then enter a new description in the target language.

6. Translate the Mobile view labels:

Note. If available, the descriptions that you can translate display next to the fields in the base language.

- a. Click the Mobile Page View Labels button. The page refreshes and the View Labels page displays.
- b. In the Detail View Label field, enter a translation for the Detail View in the target language.
- c. In the List View Label field, enter a translation for the List View in the target language.

Note. A default description in the target language may display in the field. If so, clear the field then enter a new description in the target language.

- d. Click OK.

7. Click Save.

Translating Mobile Page Items

To translate Mobile page items use the Translate Mobile Page Items page.

This section provides an overview of the Translate Mobile Page Items page and describes how to:

- Select a mobile page and mobile page item(s) to translate.
- Choose predefined field label descriptions in the target language.
- Specify static text for a field label description in the target language.
- Suppress the display of field descriptions.

Translate Mobile Page Items page

The fields with which you can work on this page are:

Mobile Page Name	Name of the mobile page that contains the items to translate.
Target Language	Select from the dropdown list the target language for the translations of the mobile page items.
Get Mobile Item	Click this button to retrieve the mobile page items for the selected mobile page name.
Parent/Item	This field is read-only. The Parent is the collection or object that contains the item. The Item is a collection or field on a page.
Base Label Text	Name of the mobile page item to be translated, in the base language.
Related Lang Short/Long/Text	Field that displays or where you enter the translation for the base label text. The translation that displays in this field is based upon your selection in the Label Type box. To the right of this field, the mobile device to which this field applies, such as laptop, PDA and so forth, displays.
View Label	Click this button to apply changes to the Related Lang Short/Long/Text field.
Related Lang Label ID	Click the Look Up button to select a different label ID and its associated long or short description.
Label Type	Determines the description of the label ID that displays in the Related Lang Short/Long/Text field when you click the View Labels button The options are: <ul style="list-style-type: none"> • None

- Text
- RFT Short
- RFT Long

Detail View Label In the target language, enter a label for the detail view of the page.

List View Label In the target language, enter a label for the list view of the page.

Tab Label In the target language, enter a label for the tab.

To select a mobile page and mobile page item(s) to translate:

1. Select PeopleTools, Translations, Translate System Definitions, Mobile Page Items. The Translate Mobile Page Item page displays.
2. In the Mobile Page Name field, enter the name of the page that contains the page items you want to translate, or use the Lookup button to search for the page.
3. From the Target Language dropdown list, select the target language.
4. Click the Get Mobile Item button.

The page refreshes and in the Mobile Page Items section, the first item associated with the Mobile page displays.

Click View All, or the Show Previous and Show Next buttons, to navigate through the results.

To choose predefined field label descriptions in the target language:

1. Select a mobile page with which to work.
2. In the Label Type box click the RFT Long button or the RFT Short button.
3. Click View Labels. The corresponding predefined RFT long or RFT short translation displays in the Related Lang Short/Long/Text field.
4. Click Save.

To specify static text for a field label description in the target language:

1. Select a mobile page and mobile page item with which to work.
2. In the Label Type box, click Text.
3. Click the View Labels button. Notice the value in the Related Lang Short/Long/Text field clears.
4. In the Related Lang Short/Long/Text field, enter a translation in the target language.
5. Click Save.

To suppress the display of a field description:

1. Select a mobile page and the mobile page item with which to work.
2. In the Label Type box click the None button.
3. Click Save.

Translating Portals

The following table lists where to translate PeopleSoft portal objects:

Object	Base Table	Related Language Table	Where to Translate
Portal Structures	PSPRSMDEFN	PSPRSMDEFNLANG	Translate Portal Objects page.
Portal Attribute Values	PSPRSMATTRVAL	PSPRSMATTRVALNG	Translate Portal Objects page.
Portal Attributes	PSPRSMATTR	PSPRSMATTRLANG	Translate Portal Objects page.

To translate portal objects:

1. Select PeopleTools, Translations, Translate System Definitions, Portal Objects. The Translate Portal Objects page displays.
The Translate Portal Objects page enables you to select a portal objects name and portal reference type and then translate the label and description of the portal into a specific target language.
2. In the Target Language field, select the language into which to translate the Portal Objects.
3. Enter the Portal Object Name and the Portal Reference Type into the appropriate fields.
Click the Search button to prompt for a list of names.
4. Click the Get Objects button to bring the Portal Objects into the definition list.
A set of portal objects appears in the Look Up List.
5. Translate the Portal Objects description and label into the target language.
Enter descriptions for the portal objects in the target language.
6. Save the page.

Translating Trees

The following table lists where to translate tree definitions:

Object	Base Table	Related Language Table	Where to Translate
Tree Definitions	PSTREEDEFN	PSTREEDEFNLANG	Tree Manager, Tree Manager.

In PeopleSoft Tree Manager, you can translate the tree description and the labels of nodes that correspond to language-sensitive record fields.

Definition	Where Displayed	Where Translated With Design Tools
Tree	Search dialogs in Tree Manager dialog boxes.	Tree definition in Tree Manager.
Tree Level	Tree Level label.	Application page accessed via edit level command in PeopleSoft Tree Manager.
Tree Detail Value	Tree Detail Value label.	Application page accessed via edit detail value command in PeopleSoft Tree Manager.
Tree node	Tree node label.	Application page accessed via edit tree node in PeopleSoft Tree Manager.

The translated tree descriptions appear in PeopleSoft Tree Manager list boxes.

Descriptive labels on tree nodes are derived from description fields in the record associated with the node. If you double-click the tree node, PeopleTools transfers you to the page where the node's properties are defined. If the record is language-sensitive (that is, if it has an associated related language table), the tree nodes' descriptive text is language-sensitive and can be translated using this page in the same way that you translate other application data.

This section discusses how to:

- Translate trees.
- Translate the tree structure description.

Translating Trees

To translate trees:

1. Ensure you have the specific language installed.
2. Log in to the PeopleSoft system in the target language.
3. Select Tree Manager, Tree Manager.
4. Open the tree whose description you want to translate.
Click on the root or children.
5. Click the Edit Data button (the pencil icon). The Tree Node page displays.
6. In the Description field, enter a translation in the target language.
7. Click OK.
8. Save the page.

Translating Tree Structure Descriptions

To translate the tree structure description:

1. Ensure you have the specific language installed.
2. Log in to the PeopleSoft system in the target language.
3. Select Tree Manager, Maintain Tree Structure.
4. Select the Structure ID to translate.
5. Open the tree whose description you want to translate.
Click on the root or children.
6. In the Description field, enter a translation in the target language.
7. Click OK.
8. Save the page.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Tree Manager, “Getting Started with PeopleSoft Tree Manager,” PeopleSoft Tree Manager Overview

Translating Records

The following table lists where to translate record definitions:

Object	Base Table	Related Language Table	Where to Translate
Record Definitions	PSRECDEFN	PSRECDEFNLANG	Translate Records page.

To translate record descriptions:

1. Select PeopleTools, Translations, Translate System Definitions, Records. The Translate Records page displays.
The Translate Records page lets you select a set of record definitions and then translate the short and long descriptions for those records into a specific target language.
2. In the Target Language field, select the language into which to translate the record descriptions.
3. In the Record field, select a set of record definitions.
Leave the Record field blank if you want to select all record definitions in the system.
If you want to work through the records alphabetically, enter the first characters of the record name in the Record field. If you want to translate a specific record, enter the record name in the Record field or use the Search button to prompt for a specific record.
4. Click the Get Record button to bring the record descriptions into the record list.
A set of record definitions appears in the Record List field based on the value you entered into the Record field.
5. Translate the short and long descriptions into the target language.
The base-language descriptions for each record appear on the left; enter the translations in the fields on the right.
6. Save the page.

Translating Queries

The following table lists where to translate queries:

Object	Base Table	Related Language Table	Where to Translate
Query Definitions	PSQRYDEFN	PSQRYDEFNLANG	Translate Queries page.
Query Fields	PSQRYFIELD	PSQRYFIELDLANG	Translate Queries page.
Query Binds	PSQRYBIND	PSQRYBINDLANG	Translate Queries page.

To translate queries using the Translate Query utility:

1. Select PeopleTools, Translations, Translate System Definition, Queries. The Translate Query page displays. The Queries page enables you to translate query descriptions, heading labels, and query prompt descriptions into a specific target language.
2. In the Target Language field, select the language into which to translate the query descriptions and labels.
3. In the Query Name field, select a query.

Leave the Query Name field blank if you want to select all the queries in the system.

If you want to work through the fields alphabetically, enter the first characters of the query name in the Query Name field. If you want to translate a specific query, enter the query name in the Query Name field, or click the Search button to prompt for a specific record.

4. Click the Get Query button to bring the query descriptions into the query list.

A set of query descriptions appears in the field Query List based on the value that you entered in the Query Name field.

5. Translate the base-language descriptions into the target language.

The base-language descriptions appear on the left; enter the translations in the fields on the right.

6. Translate query field headings and prompt names.

Click the Show Item Details button to display the Query Headings page.

By default, field headings in queries are derived from RFT Long or RFT Short field descriptions, which means that they are automatically language-sensitive (provided that the field descriptions have been translated). Query prompt names are similarly derived from the prompt table description. The Query Headings page displays any field heading labels and prompt descriptions that are *not* derived from field or table descriptions, that is, descriptions that have override text in the query definition.

Translate the description text for the field headings and prompt descriptions in the fields, and then click OK to accept the change and return to the Translate Query page.

7. Save the page.

See Also

Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Query

Translating Strings

The following table lists where to translate strings:

Object	Base Table	Related Language Table	Where to translate
Strings	STRINGS_TBL	STRINGS_LNG_TBL	Translate Strings page.

Translate Strings

*Program ID: *Language Code:

Strings in the Program Find | View All | First ◀ 1 of 6 ▶ Last

String ID: STDHDG_CO_NM	Width: 10		
String Source	Field Label ID	Default Lbl	String Text
Text		<input checked="" type="checkbox"/>	PeopleSoft
<input type="text" value="Text"/>		<input type="checkbox"/>	<input type="text"/>

Translate Strings page

Translating String Definitions into a Non-Base Language

To translate string definitions into a non-base language:

1. Select PeopleTools, Translations, Translate System Definitions, Strings. The Translate Strings page displays.
2. In the Program ID field, enter or search for a program ID.
3. From the Language Code dropdown list, select the target language.
4. Click the Find Strings button.

The page refreshes and displays the first of all strings for the program ID.

Click the View All link or use the Show Next and Show Previous arrows to navigate through the results.

5. In the String Text field, enter the a translation for the string that displays directly above the field.

There are three situations in which you can translate strings IDs.

- The String ID matches a field name in the database whose labels have been translated. In this case, you can use the Translate Strings page to choose among any of the labels of that field or static text for the string label. If none of the labels is appropriate for this context (or they are too long) then you can select Text as the label type and enter the translation. It is preferable that, whenever possible, you use field labels.
- The String ID matches a field name in the database whose labels have *not* been translated. In this case, the message Untranslated Field displays in red. This message means that there is a field in the database with the same name as the String ID, but it has not been translated, so you cannot use the translated labels for the string. In this situation, you must translate the field first and then return to the Translate Strings page and select the label that is more appropriate for the string.
- The String ID does not match a field name in the database. In this case, you may not select a label type. Text is the only option.

6. Save the page.

See Also

Chapter 8, “Understanding Global Reporting and Data Analysis Tools,” Understanding the Strings Table and Language-Sensitive Text in Reports, page 104

Translating Application Engine Descriptions

The following table lists where to translate PeopleSoft Application Engine and Application Message definitions.

Object	Base Table	Related Language Table	Where to translate
Application Engine Definitions	PSAEAPPLDEFN	PSAEAPPLLANG	Translate Application Engine Programs page.
Application Message Definitions	PSMSGDEFN	PSMSGDEFNLANG	Translate Message Definitions.

Translating Application Engine Programs

To translate Application Engine programs:

1. Select PeopleTools, Translations, Translate System Definition, Application Engine Programs. The Translate Application Engine Programs page displays.

This page enables you to translate Application Engine program descriptions into a specific target language.

2. In the Target Language field, select the language into which to translate.
3. In the Program field, select an Application Engine program.

Leave the Program name field blank if you want to select all the programs in the system.

If you want to work through the fields alphabetically, enter the first characters of the program name in the Program field. If you want to translate a specific Application Engine program, enter the program name in the Program field, or click the Search button to prompt for a specific record.

4. Click the Get Objects button to bring the program name and base descriptions into the object list.

A set of base program descriptions appears in the Object List field based on the value that you entered in the Program field.

5. Translate the base-language descriptions into the target language.

The base-language descriptions appear on the left; enter the translations in the fields on the right.

6. Save the page.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Application Engine*, “Understanding PeopleSoft Application Engine”.

Translating Message Definitions

To translate message definitions:

1. Select PeopleTools, Translations, Translate System Definition, Translate Message Definitions.

The Translate Message Definitions page displays. This page enables you to translate message definition descriptions into a specific target language.

2. In the Target Language field, select the language into which to translate.
3. In the Program field, select a message definition program.
Leave the Message name field blank if you want to select from all the programs in the system.
4. Click the Get Objects button to bring the message definition, description and long descriptions into the object list.
The message name description appears in the Object List field based on the value that you entered in the Message field.
5. Translate the base-language descriptions into the target language.
The descriptions appear on the left; enter the translations in the fields on the right.
6. Save the page.

See *Enterprise PeopleTools 8.45 PeopleBook: PeopleSoft Integration Broker*, “Defining Message Channels and Messages,” Defining Message Definitions.

Translating Review Page Text

Most page text is derived from language-sensitive field descriptions. Text derived from field descriptions is translated automatically when you clone the base-language page definition (provided that the field descriptions have already been translated). However, page control definitions give you the option to override the field description with a text description.

If you use Translation Designer to translate pages, you can translate all such override text right along with labels that are derived from field descriptions. This feature of Translation Designer helps to ensure that the entire page gets translated.

PeopleSoft also provides a Page Text inquiry page that identifies override text in pages to help you identify areas where you may have missed translating. You can translate the override text in the target-language page definition in PeopleSoft Application Designer.

To check for override text in pages:

1. Select PeopleTools, Translations, Translate System Definitions, Review Page Text. The Review Page Text page displays. The Review Page Text page lets you generate a list of all override text (that is, text that is not derived from field descriptions) in a set of page definitions.
2. In the Page Name field, select a set of page names.
Leave the Page Name field blank if you want to select all the pages in the system.
If you want to work through the pages alphabetically, enter the first characters of the page name in the Page Name field. If you want to translate a specific page, enter the page name in the Page Name field, or click the Search button to prompt for a specific record.
3. In the Language Code field, select the language code of the page set.
For example, if the Page Name field contains *B*, and the Language Code field contains *French*, the page set will consist of page definitions that begin with *B* and whose language code is *FRA*.
4. Click the Get Page Text button to display the overridden page text.
A set of page definitions appears based on the settings in the Page Name and Language Code fields.
For each page that appears in the list, the label text is the override text on the page. Looking at the label text, you can tell whether it has been translated. If it has not been translated, use Translation Designer to do the translation.

See [Chapter 15, “Translating Application Definitions,” Using the Translation Designer Feature, page 217.](#)

5. Save the page.

Translating Application Data

The Application Data page enables you to translate system and customer data. This page does not allow you to translate managed objects.

To access the Translate Application Data page, select PeopleTools, Translations, Translate System Definitions, Application Data.

Note. To use the Application Data page you must be logged into the PeopleSoft Pure Internet Architecture (PIA) in the base language.

Before you can use this feature, you must enable access to the Translate Application Data page. This section describes how to:

- Enable access to the Translate Application Data page.
- Translate record fields or ignore record fields for translation.

Enabling Access to the Translate Application Data Page

To enable access to the Translate Application Data page:

1. Select PeopleTools, Security, Permissions & Roles, Permission Lists. The Permissions List page displays.
2. Click the permission list that your user ID is using. Your permission list displays.
3. Click the Pages tab.
4. Navigate to the Translate menu name and click the Edit Components link. The Component Permissions list displays.
5. Navigate to the TRANSLATE_DATA component and click the Edit Pages link. The Page Permissions page displays.
6. Click the Select All button.
7. Click the OK button, and then click the Save button.
8. Repeat steps 2 through 7 for other groups that need access to the page.
9. Select PeopleTools, Portal, Structure and Content.
10. Click PeopleTools.
11. Click Translate.
12. Click Translate System Definitions.
13. Click Edit for the Application Data row.
14. Clear the Hide in Navigation check box.
15. Save your changes.

Translating Application Data

To translate application data:

1. Select PeopleTools, Translations, Translate System Definitions, Application Data. The Translate Application Data page displays.
2. In the Record to Translate field, enter the base table name.
3. From the Target Language Code dropdown list, select the target language.
4. Select the record fields to translate and click Search.

When you click Search, the page displays the related language record, the number of rows or values found, and the first of all key names and key values for the table. For each value, the translatable fields display in the Base Text column in the base language of the database. The Lang Text fields display translations for the fields, if they exist. Use the View All link or the Show Previous and Show Next arrows to navigate through the results retrieved.

Note. The following information describes how to work with all rows in a record or specific rows in a record.

- To work with all rows in a record, click Search.
- To select specific rows in a record, click the Filter with SQL? link.

A WHERE clause page displays that provides you with an area to enter an SQL statement that will filter the record data. Enter a WHERE clause to filter the data.

The field names that you can use in the WHERE clause display in the Record Fields box. To expand the list, click the View All link.

After you enter a WHERE clause, click OK to return to the Translate Application Data page. Click Search.

5. Translate or ignore the record fields.

Some related language tables contain fields that should never be translated. You can choose to translate the record fields or tell the system to ignore them for translation.

- To translate a field, in the Lang Text box, enter a translation for the item in the Base Text field.
- To ignore a field for translation, check the Ignore? box next to the Lang Text box. The following message displays:

```
Ignoring field: Translations for this field will be overwritten with base=>
language values, proceed? (102,83)
```

```
By ignoring this field, any translation for this field will be overwritten with=>
the base language values. Do you want to proceed?
```

Click Yes.

Note. This message displays the first time you check the Ignore? box for a translatable field. If you check this box again for the same field, the message will not display and the system copies, in the base language, the text in the Base Text field over to the Lang Text field. This text will appear in all rows in the table. Therefore, do not use the Ignore? option to simply copy the information in the Base Text field to the Lang Text box. To remove an entry in the Lang Text box made under these circumstances, clear the Ignore? box and delete the entry.

6. Click Save.

Translating HTML Definitions

You typically translate HTML definitions when they contain hard-coded text. Most HTML definitions should contain JavaScript and references to the Message Catalog, only and therefore don't require separate translation, however it is possible to hardcode text into HTML definitions which may require translation using this method.

To translate an HTML definition:

1. Change the PeopleSoft Configuration Manager language preference to the target language.
2. Login to the PeopleSoft system.
3. Open PeopleSoft Application Designer.
4. Open the HTML definition.

Choose File, Open and select the HTML definition you want to open.

5. Edit any language-sensitive text in the HTML object.
6. Save the HTML object.

APPENDIX A

ISO Country and Currency Codes

PeopleBooks use International Organization for Standardization (ISO) country and currency codes to identify country-specific information and monetary amounts.

This appendix discusses:

- ISO country codes.
- ISO currency codes.

See Also

"About This PeopleBook." Typographical Conventions and Visual Cues

ISO Country Codes

This table lists the ISO country codes that may appear as country identifiers in PeopleBooks:

ISO Country Code	Country Name
ABW	Aruba
AFG	Afghanistan
AGO	Angola
AIA	Anguilla
ALB	Albania
AND	Andorra
ANT	Netherlands Antilles
ARE	United Arab Emirates
ARG	Argentina
ARM	Armenia
ASM	American Samoa
ATA	Antarctica

ISO Country Code	Country Name
ATF	French Southern Territories
ATG	Antigua and Barbuda
AUS	Australia
AUT	Austria
AZE	Azerbaijan
BDI	Burundi
BEL	Belgium
BEN	Benin
BFA	Burkina Faso
BGD	Bangladesh
BGR	Bulgaria
BHR	Bahrain
BHS	Bahamas
BIH	Bosnia and Herzegovina
BLR	Belarus
BLZ	Belize
BMU	Bermuda
BOL	Bolivia
BRA	Brazil
BRB	Barbados
BRN	Brunei Darussalam
BTN	Bhutan
BVT	Bouvet Island
BWA	Botswana
CAF	Central African Republic
CAN	Canada
CCK	Cocos (Keeling) Islands

ISO Country Code	Country Name
CHE	Switzerland
CHL	Chile
CHN	China
CIV	Cote D'Ivoire
CMR	Cameroon
COD	Congo, The Democratic Republic
COG	Congo
COK	Cook Islands
COL	Colombia
COM	Comoros
CPV	Cape Verde
CRI	Costa Rica
CUB	Cuba
CXR	Christmas Island
CYM	Cayman Islands
CYP	Cyprus
CZE	Czech Republic
DEU	Germany
DJI	Djibouti
DMA	Dominica
DNK	Denmark
DOM	Dominican Republic
DZA	Algeria
ECU	Ecuador
EGY	Egypt
ERI	Eritrea
ESH	Western Sahara

ISO Country Code	Country Name
ESP	Spain
EST	Estonia
ETH	Ethiopia
FIN	Finland
FJI	Fiji
FLK	Falkland Islands (Malvinas)
FRA	France
FRO	Faroe Islands
FSM	Micronesia, Federated States
GAB	Gabon
GBR	United Kingdom
GEO	Georgia
GHA	Ghana
GIB	Gibraltar
GIN	Guinea
GLP	Guadeloupe
GMB	Gambia
GNB	Guinea-Bissau
GNQ	Equatorial Guinea
GRC	Greece
GRD	Grenada
GRL	Greenland
GTM	Guatemala
GUF	French Guiana
GUM	Guam
GUY	Guyana
GXA	GXA - GP Core Country

ISO Country Code	Country Name
GXB	GXB - GP Core Country
GXC	GXC - GP Core Country
GXD	GXD - GP Core Country
HKG	Hong Kong
HMD	Heard and McDonald Islands
HND	Honduras
HRV	Croatia
HTI	Haiti
HUN	Hungary
IDN	Indonesia
IND	India
IOT	British Indian Ocean Territory
IRL	Ireland
IRN	Iran (Islamic Republic Of)
IRQ	Iraq
ISL	Iceland
ISR	Israel
ITA	Italy
JAM	Jamaica
JOR	Jordan
JPN	Japan
KAZ	Kazakstan
KEN	Kenya
KGZ	Kyrgyzstan
KHM	Cambodia
KIR	Kiribati
KNA	Saint Kitts and Nevis

ISO Country Code	Country Name
KOR	Korea, Republic of
KWT	Kuwait
LAO	Lao People's Democratic Rep
LBN	Lebanon
LBR	Liberia
LBY	Libyan Arab Jamahiriya
LCA	Saint Lucia
LIE	Liechtenstein
LKA	Sri Lanka
LSO	Lesotho
LTU	Lithuania
LUX	Luxembourg
LVA	Latvia
MAC	Macao
MAR	Morocco
MCO	Monaco
MDA	Moldova, Republic of
MDG	Madagascar
MDV	Maldives
MEX	Mexico
MHL	Marshall Islands
MKD	Fmr Yugoslav Rep of Macedonia
MLI	Mali
MLT	Malta
MMR	Myanmar
MNG	Mongolia
MNP	Northern Mariana Islands

ISO Country Code	Country Name
MOZ	Mozambique
MRT	Mauritania
MSR	Montserrat
MTQ	Martinique
MUS	Mauritius
MWI	Malawi
MYS	Malaysia
MYT	Mayotte
NAM	Namibia
NCL	New Caledonia
NER	Niger
NFK	Norfolk Island
NGA	Nigeria
NIC	Nicaragua
NIU	Niue
NLD	Netherlands
NOR	Norway
NPL	Nepal
NRU	Nauru
NZL	New Zealand
OMN	Oman
PAK	Pakistan
PAN	Panama
PCN	Pitcairn
PER	Peru
PHL	Philippines
PLW	Palau

ISO Country Code	Country Name
PNG	Papua New Guinea
POL	Poland
PRI	Puerto Rico
PRK	Korea, Democratic People's Rep
PRT	Portugal
PRY	Paraguay
PSE	Palestinian Territory, Occupie
PYF	French Polynesia
QAT	Qatar
REU	Reunion
ROU	Romania
RUS	Russian Federation
RWA	Rwanda
SAU	Saudi Arabia
SDN	Sudan
SEN	Senegal
SGP	Singapore
SGS	Sth Georgia & Sth Sandwich Is
SHN	Saint Helena
SJM	Svalbard and Jan Mayen
SLB	Solomon Islands
SLE	Sierra Leone
SLV	El Salvador
SMR	San Marino
SOM	Somalia
SPM	Saint Pierre and Miquelon
STP	Sao Tome and Principe

ISO Country Code	Country Name
SUR	Suriname
SVK	Slovakia
SVN	Slovenia
SWE	Sweden
SWZ	Swaziland
SYC	Seychelles
SYR	Syrian Arab Republic
TCA	Turks and Caicos Islands
TCD	Chad
TGO	Togo
THA	Thailand
TJK	Tajikistan
TKL	Tokelau
TKM	Turkmenistan
TLS	East Timor
TON	Tonga
TTO	Trinidad and Tobago
TUN	Tunisia
TUR	Turkey
TUV	Tuvalu
TWN	Taiwan, Province of China
TZA	Tanzania, United Republic of
UGA	Uganda
UKR	Ukraine
UMI	US Minor Outlying Islands
URY	Uruguay
USA	United States

ISO Country Code	Country Name
UZB	Uzbekistan
VAT	Holy See (Vatican City State)
VCT	St Vincent and the Grenadines
VEN	Venezuela
VGB	Virgin Islands (British)
VIR	Virgin Islands (U.S.)
VNM	Viet Nam
VUT	Vanuatu
WLF	Wallis and Futuna Islands
WSM	Samoa
YEM	Yemen
YUG	Yugoslavia
ZAF	South Africa
ZMB	Zambia
ZWE	Zimbabwe

ISO Currency Codes

This table lists the ISO country codes that may appear as currency identifiers in PeopleBooks:

ISO Currency Code	Description
ADP	Andorran Peseta
AED	United Arab Emirates Dirham
AFA	Afghani
AFN	Afghani
ALK	Old Lek
ALL	Lek
AMD	Armenian Dram

ISO Currency Code	Description
ANG	Netherlands Antilles Guilder
AOA	Kwanza
AOK	Kwanza
AON	New Kwanza
AOR	Kwanza Reajustado
ARA	Austral
ARP	Peso Argentino
ARS	Argentine Peso
ARY	Peso
ATS	Schilling
AUD	Australian Dollar
AWG	Aruban Guilder
AZM	Azerbaijani Manat
BAD	Dinar
BAM	Convertible Marks
BBD	Barbados Dollar
BDT	Taka
BEC	Convertible Franc
BEF	Belgian Franc
BEL	Financial Belgian Franc
BGJ	Lev A/52
BGK	Lev A/62
BGL	Lev
BGN	Bulgarian LEV
BHD	Bahraini Dinar
BIF	Burundi Franc
BMD	Bermudian Dollar

ISO Currency Code	Description
BND	Brunei Dollar
BOB	Boliviano
BOP	Peso
BOV	Mvdol
BRB	Cruzeiro
BRC	Cruzado
BRE	Cruzeiro
BRL	Brazilian Real
BRN	New Cruzado
BRR	Brazilian Real Dollar
BSD	Bahamian Dollar
BTN	Ngultrum
BUK	N/A
BWP	Pula
BYB	Belarussian Ruble
BYR	Belarussian Ruble
BZD	Belize Dollar
CAD	Canadian Dollar
CDF	Franc Congolais
CHF	Swiss Franc
CLF	Unidades de fomento
CLP	Chilean Peso
CNX	Peoples Bank Dollar
CNY	Yuan Renminbi
COP	Colombian Peso
CRC	Costa Rican Colon
CSD	Serbia Dinar

ISO Currency Code	Description
CSJ	Krona A/53
CSK	Koruna
CUP	Cuban Peso
CVE	Cape Verde Escudo
CYP	Cyprus Pound
CZK	Czech Koruna
DEM	Deutsche Mark
DJF	Djibouti Franc
DKK	Danish Krone
DOP	Dominican Peso
DZD	Algerian Dinar
ECS	Sucre
ECV	Unidad de Valor
EEK	Kroon
EGP	Egyptian Pound
EQE	Ekwele
ERN	Nakfa
ESA	Spanish Peseta
ESB	Convertible Peseta
ESP	Spanish Peseta
ETB	Ethiopian Birr
EUR	euro
FIM	Markka
FJD	Fiji Dollar
FKP	Falklands Isl. Pound
FRF	French Franc
GBP	Pound Sterling

ISO Currency Code	Description
GEK	Georgian Coupon
GEL	Lari
GHC	Cedi
GIP	Gibraltar Pound
GMD	Dalasi
GNE	Syli
GNF	Guinea Franc
GNS	Syli
GQE	Ekwele
GRD	Drachma
GTQ	Quetzal
GWE	Guinea Escudo
GWP	Guinea-Bissau Peso
GYD	Guyana Dollar
HKD	Hong Kong Dollar
HNL	Lempira
HRD	Dinar
HRK	Kuna
HTG	Gourde
HUF	Forint
IDR	Rupiah
IEP	Irish Pound
ILP	Pound
ILR	Old Shekel
ILS	New Israeli Sheqel
INR	Indian Rupee
IQD	Iraqi Dinar

ISO Currency Code	Description
IRR	Iranian Rial
ISJ	Old Krona
ISK	Iceland Krona
ITL	Italian Lira
JMD	Jamaican Dollar
JOD	Jordanian Dinar
JPY	Yen
KES	Kenyan Shilling
KGS	Som
KHR	Riel
KMF	Comoro Franc
KPW	North Korean Won
KRW	Won
KWD	Kuwaiti Dinar
KYD	Cayman Islands dollar
KZT	Tenge
LAJ	Kip Pot Pol
LAK	Kip
LBP	Lebanese Pound
LKR	Sri Lanka Rupee
LRD	Liberian Dollar
LSL	Loti
LSM	Maloti
LTL	Lithuanian Litas
LTT	Talonas
LUC	Convertib Franc
LUF	Luxembourg Franc

ISO Currency Code	Description
LUL	Financial Franc
LVL	Latvian Lats
LVR	Latvian Ruble
LYD	Libyan Dinar
MAD	Moroccan Dirham
MAF	Mali Franc
MDL	Moldovan Leu
MGF	Malagasy Franc
MKD	Denar
MLF	Mali Franc
MMK	Kyat
MNT	Tugrik
MOP	Pataca
MRO	Ouguiya
MTL	Maltese Lira
MTP	Maltese Pound
MUR	Mauritius Rupee
MVQ	Maldives Rupee
MVR	Rufiyaa
MWK	Malawian Kwacha
MXN	Mexican Peso
MXP	Mexican Peso
MXV	Mexican UDI
MYR	Malaysian Ringgit
MZE	Mozambique Escudo
MZM	Metical
NAD	Namibia Dollar

ISO Currency Code	Description
NGN	Naira
NIC	Cordoba
NIO	Cordoba Oro
NLG	Netherlands Guilder
NOK	Norwegian Krone
NPR	Nepalese Rupee
NZD	New Zealand Dollar
OMR	Rial Omani
PAB	Balboa
PEI	Inti
PEN	Nuevo Sol
PES	Sol
PGK	Kina
PHP	Philippine Peso
PKR	Pakistan Rupee
PLN	Zloty
PLZ	Zloty
PTE	Portuguese Escudo
PYG	Guarani
QAR	Qatari Rial
ROK	Leu A/52
ROL	Leu
RUB	Russian Ruble
RUR	Russian Federation Rouble
RWF	Rwanda Franc
SAR	Saudi Riyal
SBD	Solomon Islands

ISO Currency Code	Description
SCR	Seychelles Rupee
SDD	Sudanese Dinar
SDP	Sudanese Pound
SEK	Swedish Krona
SGD	Singapore Dollar
SHP	St Helena Pound
SIT	Tolar
SKK	Slovak Koruna
SLL	Leone
SOS	Somali Shilling
SRG	Surinam Guilder
STD	Dobra
SUR	Rouble
SVC	El Salvador Colon
SYP	Syrian Pound
SZL	Lilangeni
THB	Baht
TJR	Tajik Ruble
TJS	Somoni
TMM	Manat
TND	Tunisian Dinar
TOP	Pa'anga
TPE	Timor Escudo
TRL	Turkish Lira
TTD	Trinidad Dollar
TWD	New Taiwan Dollar
TZS	Tanzanian Shilling

ISO Currency Code	Description
UAH	Hryvnia
UAK	Karbovanet
UGS	Uganda Shilling
UGW	Old Shilling
UGX	Uganda Shilling
USD	US Dollar
USN	US Dollar (Next day)
USS	US Dollar (Same day)
UYN	Old Uruguay Peso
UYP	Uruguayan Peso
UYU	Peso Uruguayo
UZS	Uzbekistan Sum
VEB	Bolivar
VNC	Old Dong
VND	Dong
VUV	Vatu
WST	Tala
XAF	CFA Franc BEAC
XAG	Silver
XAU	GOLD
XBA	European Composite Unit
XBB	European Monetary Unit
XBC	European Unit of Account 9
XBD	European Unit of Account 17
XCD	East Caribbean Dollar
XDR	SDR
XEU	EU Currency (E.C.U)

ISO Currency Code	Description
XFO	Gold-Franc
XFU	UIC-Franc
XOF	CFA Franc BCEAO
XPD	Palladium
XPF	CFP Franc
XPT	Platinum
XTS	For Testing Purposes
XXX	Non Currency Transaction
YDD	Yemeni Din
YER	Yemeni Rial
YUD	New Yugoslavian Dinar
YUM	New Dinar
YUN	Yugoslavian Dinar
ZAL	Financial Rand
ZAR	Rand
ZMK	Zambian Kwacha
ZRN	New Zaire
ZRZ	Zaire
ZWC	Rhodesian Dollar
ZWD	Zimbabwe Dollar

Glossary of PeopleSoft Terms

absence entitlement	This element defines rules for granting paid time off for valid absences, such as sick time, vacation, and maternity leave. An absence entitlement element defines the entitlement amount, frequency, and entitlement period.
absence take	This element defines the conditions that must be met before a payee is entitled to take paid time off.
accounting class	In PeopleSoft Enterprise Performance Management, the accounting class defines how a resource is treated for generally accepted accounting practices. The Inventory class indicates whether a resource becomes part of a balance sheet account, such as inventory or fixed assets, while the Non-inventory class indicates that the resource is treated as an expense of the period during which it occurs.
accounting date	The accounting date indicates when a transaction is recognized, as opposed to the date the transaction actually occurred. The accounting date and transaction date can be the same. The accounting date determines the period in the general ledger to which the transaction is to be posted. You can only select an accounting date that falls within an open period in the ledger to which you are posting. The accounting date for an item is normally the invoice date.
accounting split	The accounting split method indicates how expenses are allocated or divided among one or more sets of accounting ChartFields.
accumulator	You use an accumulator to store cumulative values of defined items as they are processed. You can accumulate a single value over time or multiple values over time. For example, an accumulator could consist of all voluntary deductions, or all company deductions, enabling you to accumulate amounts. It allows total flexibility for time periods and values accumulated.
action reason	The reason an employee's job or employment information is updated. The action reason is entered in two parts: a personnel action, such as a promotion, termination, or change from one pay group to another—and a reason for that action. Action reasons are used by PeopleSoft Human Resources, PeopleSoft Benefits Administration, PeopleSoft Stock Administration, and the COBRA Administration feature of the Base Benefits business process.
action template	In PeopleSoft Receivables, outlines a set of escalating actions that the system or user performs based on the period of time that a customer or item has been in an action plan for a specific condition.
activity	<p>In PeopleSoft Enterprise Learning Management, an instance of a catalog item (sometimes called a class) that is available for enrollment. The activity defines such things as the costs that are associated with the offering, enrollment limits and deadlines, and waitlisting capacities.</p> <p>In PeopleSoft Enterprise Performance Management, the work of an organization and the aggregation of actions that are used for activity-based costing.</p> <p>In PeopleSoft Project Costing, the unit of work that provides a further breakdown of projects—usually into specific tasks.</p> <p>In PeopleSoft Workflow, a specific transaction that you might need to perform in a business process. Because it consists of the steps that are used to perform a transaction, it is also known as a step map.</p>

agreement	In PeopleSoft eSettlements, provides a way to group and specify processing options, such as payment terms, pay from a bank, and notifications by a buyer and supplier location combination.
allocation rule	In PeopleSoft Enterprise Incentive Management, an expression within compensation plans that enables the system to assign transactions to nodes and participants. During transaction allocation, the allocation engine traverses the compensation structure from the current node to the root node, checking each node for plans that contain allocation rules.
alternate account	A feature in PeopleSoft General Ledger that enables you to create a statutory chart of accounts and enter statutory account transactions at the detail transaction level, as required for recording and reporting by some national governments.
AR specialist	Abbreviation for <i>receivables specialist</i> . In PeopleSoft Receivables, an individual in who tracks and resolves deductions and disputed items.
arbitration plan	In PeopleSoft Enterprise Pricer, defines how price rules are to be applied to the base price when the transaction is priced.
assessment rule	In PeopleSoft Receivables, a user-defined rule that the system uses to evaluate the condition of a customer's account or of individual items to determine whether to generate a follow-up action.
asset class	An asset group used for reporting purposes. It can be used in conjunction with the asset category to refine asset classification.
attribute/value pair	In PeopleSoft Directory Interface, relates the data that makes up an entry in the directory information tree.
authentication server	A server that is set up to verify users of the system.
base time period	In PeopleSoft Business Planning, the lowest level time period in a calendar.
benchmark job	In PeopleSoft Workforce Analytics, a benchmark job is a job code for which there is corresponding salary survey data from published, third-party sources.
book	In PeopleSoft Asset Management, used for storing financial and tax information, such as costs, depreciation attributes, and retirement information on assets.
branch	A tree node that rolls up to nodes above it in the hierarchy, as defined in PeopleSoft Tree Manager.
budgetary account only	An account used by the system only and not by users; this type of account does not accept transactions. You can only budget with this account. Formerly called "system-maintained account."
budget check	In commitment control, the processing of source transactions against control budget ledgers, to see if they pass, fail, or pass with a warning.
budget control	In commitment control, budget control ensures that commitments and expenditures don't exceed budgets. It enables you to track transactions against corresponding budgets and terminate a document's cycle if the defined budget conditions are not met. For example, you can prevent a purchase order from being dispatched to a vendor if there are insufficient funds in the related budget to support it.
budget period	The interval of time (such as 12 months or 4 quarters) into which a period is divided for budgetary and reporting purposes. The ChartField allows maximum flexibility to define operational accounting time periods without restriction to only one calendar.
business event	In PeopleSoft Receivables, defines the processing characteristics for the Receivable Update process for a draft activity.

	In PeopleSoft Sales Incentive Management, an original business transaction or activity that may justify the creation of a PeopleSoft Enterprise Incentive Management event (a sale, for example).
business unit	A corporation or a subset of a corporation that is independent with regard to one or more operational or accounting functions.
buyer	In PeopleSoft eSettlements, an organization (or business unit, as opposed to an individual) that transacts with suppliers (vendors) within the system. A buyer creates payments for purchases that are made in the system.
catalog item	In PeopleSoft Enterprise Learning Management, a specific topic that a learner can study and have tracked. For example, "Introduction to Microsoft Word." A catalog item contains general information about the topic and includes a course code, description, categorization, keywords, and delivery methods. A catalog item can have one or more learning activities.
catalog map	In PeopleSoft Catalog Management, translates values from the catalog source data to the format of the company's catalog.
catalog partner	In PeopleSoft Catalog Management, shares responsibility with the enterprise catalog manager for maintaining catalog content.
categorization	Associates partner offerings with catalog offerings and groups them into enterprise catalog categories.
channel	In PeopleSoft MultiChannel Framework, email, chat, voice (computer telephone integration [CTI]), or a generic event.
ChartField	A field that stores a chart of accounts, resources, and so on, depending on the PeopleSoft application. ChartField values represent individual account numbers, department codes, and so forth.
ChartField balancing	You can require specific ChartFields to match up (balance) on the debit and the credit side of a transaction.
ChartField combination edit	The process of editing journal lines for valid ChartField combinations based on user-defined rules.
ChartKey	One or more fields that uniquely identify each row in a table. Some tables contain only one field as the key, while others require a combination.
checkbook	In PeopleSoft Promotions Management, enables you to view financial data (such as planned, incurred, and actual amounts) that is related to funds and trade promotions.
Class ChartField	A ChartField value that identifies a unique appropriation budget key when you combine it with a fund, department ID, and program code, as well as a budget period. Formerly called <i>sub-classification</i> .
clone	In PeopleCode, to make a unique copy. In contrast, to <i>copy</i> may mean making a new reference to an object, so if the underlying object is changed, both the copy and the original change.
collection	To make a set of documents available for searching in Verity, you must first create at least one collection. A collection is set of directories and files that allow search application users to use the Verity search engine to quickly find and display source documents that match search criteria. A collection is a set of statistics and pointers to the source documents, stored in a proprietary format on a file server. Because a collection can only store information for a single location, PeopleSoft maintains a set of collections (one per language code) for each search index object.

collection rule	In PeopleSoft Receivables, a user-defined rule that defines actions to take for a customer based on both the amount and the number of days past due for outstanding balances.
compensation object	In PeopleSoft Enterprise Incentive Management, a node within a compensation structure. Compensation objects are the building blocks that make up a compensation structure's hierarchical representation.
compensation structure	In PeopleSoft Enterprise Incentive Management, a hierarchical relationship of compensation objects that represents the compensation-related relationship between the objects.
condition	In PeopleSoft Receivables, occurs when there is a change of status for a customer's account, such as reaching a credit limit or exceeding a user-defined balance due.
configuration parameter catalog	Used to configure an external system with PeopleSoft. For example, a configuration parameter catalog might set up configuration and communication parameters for an external server.
configuration plan	In PeopleSoft Enterprise Incentive Management, configuration plans hold allocation information for common variables (not incentive rules) and are attached to a node without a participant. Configuration plans are not processed by transactions.
content reference	Content references are pointers to content registered in the portal registry. These are typically either URLs or iScripts. Content references fall into three categories: target content, templates, and template pagelets.
context	In PeopleCode, determines which buffer fields can be contextually referenced and which is the current row of data on each scroll level when a PeopleCode program is running. In PeopleSoft Enterprise Incentive Management, a mechanism that is used to determine the scope of a processing run. PeopleSoft Enterprise Incentive Management uses three types of context: plan, period, and run-level.
control table	Stores information that controls the processing of an application. This type of processing might be consistent throughout an organization, or it might be used only by portions of the organization for more limited sharing of data.
cost profile	A combination of a receipt cost method, a cost flow, and a deplete cost method. A profile is associated with a cost book and determines how items in that book are valued, as well as how the material movement of the item is valued for the book.
cost row	A cost transaction and amount for a set of ChartFields.
current learning	In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's in-progress learning activities and programs.
data acquisition	In PeopleSoft Enterprise Incentive Management, the process during which raw business transactions are acquired from external source systems and fed into the operational data store (ODS).
data elements	Data elements, at their simplest level, define a subset of data and the rules by which to group them. For Workforce Analytics, data elements are rules that tell the system what measures to retrieve about your workforce groups.
dataset	A data grouping that enables role-based filtering and distribution of data. You can limit the range and quantity of data that is displayed for a user by associating dataset rules with user roles. The result of dataset rules is a set of data that is appropriate for the user's roles.

delivery method	<p>In PeopleSoft Enterprise Learning Management, identifies the primary type of delivery method in which a particular learning activity is offered. Also provides default values for the learning activity, such as cost and language. This is primarily used to help learners search the catalog for the type of delivery from which they learn best. Because PeopleSoft Enterprise Learning Management is a blended learning system, it does not enforce the delivery method.</p> <p>In PeopleSoft Supply Chain Management, identifies the method by which goods are shipped to their destinations (such as truck, air, rail, and so on). The delivery method is specified when creating shipment schedules.</p>
delivery method type	In PeopleSoft Enterprise Learning Management, identifies how learning activities can be delivered—for example, through online learning, classroom instruction, seminars, books, and so forth—in an organization. The type determines whether the delivery method includes scheduled components.
directory information tree	In PeopleSoft Directory Interface, the representation of a directory's hierarchical structure.
document sequencing	A flexible method that sequentially numbers the financial transactions (for example, bills, purchase orders, invoices, and payments) in the system for statutory reporting and for tracking commercial transaction activity.
dynamic detail tree	A tree that takes its detail values—dynamic details—directly from a table in the database, rather than from a range of values that are entered by the user.
edit table	A table in the database that has its own record definition, such as the Department table. As fields are entered into a PeopleSoft application, they can be validated against an edit table to ensure data integrity throughout the system.
effective date	A method of dating information in PeopleSoft applications. You can predate information to add historical data to your system, or postdate information in order to enter it before it actually goes into effect. By using effective dates, you don't delete values; you enter a new value with a current effective date.
EIM ledger	Abbreviation for <i>Enterprise Incentive Management ledger</i> . In PeopleSoft Enterprise Incentive Management, an object to handle incremental result gathering within the scope of a participant. The ledger captures a result set with all of the appropriate traces to the data origin and to the processing steps of which it is a result.
elimination set	In PeopleSoft General Ledger, a related group of intercompany accounts that is processed during consolidations.
entry event	In PeopleSoft General Ledger, Receivables, Payables, Purchasing, and Billing, a business process that generates multiple debits and credits resulting from single transactions to produce standard, supplemental accounting entries.
equitization	In PeopleSoft General Ledger, a business process that enables parent companies to calculate the net income of subsidiaries on a monthly basis and adjust that amount to increase the investment amount and equity income amount before performing consolidations.
event	<p>A predefined point either in the Component Processor flow or in the program flow. As each point is encountered, the event activates each component, triggering any PeopleCode program that is associated with that component and that event. Examples of events are FieldChange, SavePreChange, and RowDelete.</p> <p>In PeopleSoft Human Resources, also refers to an incident that affects benefits eligibility.</p>
event propagation process	In PeopleSoft Sales Incentive Management, a process that determines, through logic, the propagation of an original PeopleSoft Enterprise Incentive Management event and creates a derivative (duplicate) of the original event to be processed by other objects.

	Sales Incentive Management uses this mechanism to implement splits, roll-ups, and so on. Event propagation determines who receives the credit.
exception	In PeopleSoft Receivables, an item that either is a deduction or is in dispute.
exclusive pricing	In PeopleSoft Order Management, a type of arbitration plan that is associated with a price rule. Exclusive pricing is used to price sales order transactions.
fact	In PeopleSoft applications, facts are numeric data values from fields from a source database as well as an analytic application. A fact can be anything you want to measure your business by, for example, revenue, actual, budget data, or sales numbers. A fact is stored on a fact table.
forecast item	A logical entity with a unique set of descriptive demand and forecast data that is used as the basis to forecast demand. You create forecast items for a wide range of uses, but they ultimately represent things that you buy, sell, or use in your organization and for which you require a predictable usage.
fund	In PeopleSoft Promotions Management, a budget that can be used to fund promotional activity. There are four funding methods: top down, fixed accrual, rolling accrual, and zero-based accrual.
generic process type	In PeopleSoft Process Scheduler, process types are identified by a generic process type. For example, the generic process type SQR includes all SQR process types, such as SQR process and SQR report.
group	In PeopleSoft Billing and Receivables, a posting entity that comprises one or more transactions (items, deposits, payments, transfers, matches, or write-offs). In PeopleSoft Human Resources Management and Supply Chain Management, any set of records that are associated under a single name or variable to run calculations in PeopleSoft business processes. In PeopleSoft Time and Labor, for example, employees are placed in groups for time reporting purposes.
incentive object	In PeopleSoft Enterprise Incentive Management, the incentive-related objects that define and support the PeopleSoft Enterprise Incentive Management calculation process and results, such as plan templates, plans, results data, user interaction objects, and so on.
incentive rule	In PeopleSoft Sales Incentive Management, the commands that act on transactions and turn them into compensation. A rule is one part in the process of turning a transaction into compensation.
incur	In PeopleSoft Promotions Management, to become liable for a promotional payment. In other words, you owe that amount to a customer for promotional activities.
item	In PeopleSoft Inventory, a tangible commodity that is stored in a business unit (shipped from a warehouse). In PeopleSoft Demand Planning, Inventory Policy Planning, and Supply Planning, a noninventory item that is designated as being used for planning purposes only. It can represent a family or group of inventory items. It can have a planning bill of material (BOM) or planning routing, and it can exist as a component on a planning BOM. A planning item cannot be specified on a production or engineering BOM or routing, and it cannot be used as a component in a production. The quantity on hand will never be maintained.
KPI	In PeopleSoft Receivables, an individual receivable. An item can be an invoice, a credit memo, a debit memo, a write-off, or an adjustment. An abbreviation for <i>key performance indicator</i> . A high-level measurement of how well an organization is doing in achieving critical success factors. This defines the data value or calculation upon which an assessment is determined.

LDIF file	Abbreviation for <i>Lightweight Directory Access Protocol (LDAP) Data Interchange Format file</i> . Contains discrepancies between PeopleSoft data and directory data.
learner group	In PeopleSoft Enterprise Learning Management, a group of learners who are linked to the same learning environment. Members of the learner group can share the same attributes, such as the same department or job code. Learner groups are used to control access to and enrollment in learning activities and programs. They are also used to perform group enrollments and mass enrollments in the back office.
learning components	In PeopleSoft Enterprise Learning Management, the foundational building blocks of learning activities. PeopleSoft Enterprise Learning Management supports six basic types of learning components: web-based, session, webcast, test, survey, and assignment. One or more of these learning component types compose a single learning activity.
learning environment	In PeopleSoft Enterprise Learning Management, identifies a set of categories and catalog items that can be made available to learner groups. Also defines the default values that are assigned to the learning activities and programs that are created within a particular learning environment. Learning environments provide a way to partition the catalog so that learners see only those items that are relevant to them.
learning history	In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's completed learning activities and programs.
ledger mapping	You use ledger mapping to relate expense data from general ledger accounts to resource objects. Multiple ledger line items can be mapped to one or more resource IDs. You can also use ledger mapping to map dollar amounts (referred to as <i>rates</i>) to business units. You can map the amounts in two different ways: an actual amount that represents actual costs of the accounting period, or a budgeted amount that can be used to calculate the capacity rates as well as budgeted model results. In PeopleSoft Enterprise Warehouse, you can map general ledger accounts to the EW Ledger table.
library section	In PeopleSoft Enterprise Incentive Management, a section that is defined in a plan (or template) and that is available for other plans to share. Changes to a library section are reflected in all plans that use it.
linked section	In PeopleSoft Enterprise Incentive Management, a section that is defined in a plan template but appears in a plan. Changes to linked sections propagate to plans using that section.
linked variable	In PeopleSoft Enterprise Incentive Management, a variable that is defined and maintained in a plan template and that also appears in a plan. Changes to linked variables propagate to plans using that variable.
load	In PeopleSoft Inventory, identifies a group of goods that are shipped together. Load management is a feature of PeopleSoft Inventory that is used to track the weight, the volume, and the destination of a shipment.
local functionality	In PeopleSoft HRMS, the set of information that is available for a specific country. You can access this information when you click the appropriate country flag in the global window, or when you access it by a local country menu.
location	Locations enable you to indicate the different types of addresses—for a company, for example, one address to receive bills, another for shipping, a third for postal deliveries, and a separate street address. Each address has a different location number. The primary location—indicated by a <i>1</i> —is the address you use most often and may be different from the main address.
logistical task	In PeopleSoft Services Procurement, an administrative task that is related to hiring a service provider. Logistical tasks are linked to the service type on the work order so that different types of services can have different logistical tasks. Logistical tasks include both preapproval tasks (such as assigning a new badge or ordering a new

laptop) and postapproval tasks (such as scheduling orientation or setting up the service provider email). The logistical tasks can be mandatory or optional. Mandatory preapproval tasks must be completed before the work order is approved. Mandatory postapproval tasks, on the other hand, must be completed before a work order is released to a service provider.

market template	In PeopleSoft Enterprise Incentive Management, additional functionality that is specific to a given market or industry and is built on top of a product category.
match group	In PeopleSoft Receivables, a group of receivables items and matching offset items. The system creates match groups by using user-defined matching criteria for selected field values.
MCF server	Abbreviation for <i>PeopleSoft MultiChannel Framework server</i> . Comprises the universal queue server and the MCF log server. Both processes are started when <i>MCF Servers</i> is selected in an application server domain configuration.
merchandising activity	In PeopleSoft Promotions Management, a specific discount type that is associated with a trade promotion (such as off-invoice, billback or rebate, or lump-sum payment) that defines the performance that is required to receive the discount. In the industry, you may know this as an offer, a discount, a merchandising event, an event, or a tactic.
meta-SQL	Meta-SQL constructs expand into platform-specific Structured Query Language (SQL) substrings. They are used in functions that pass SQL strings, such as in SQL objects, the SQLExec function, and PeopleSoft Application Engine programs.
metastring	Metastings are special expressions included in SQL string literals. The metastings, prefixed with a percent (%) symbol, are included directly in the string literals. They expand at run time into an appropriate substring for the current database platform.
multibook	In PeopleSoft General Ledger, multiple ledgers having multiple-base currencies that are defined for a business unit, with the option to post a single transaction to all base currencies (all ledgers) or to only one of those base currencies (ledgers).
multicurrency	The ability to process transactions in a currency other than the business unit's base currency.
national allowance	In PeopleSoft Promotions Management, a promotion at the corporate level that is funded by nondiscretionary dollars. In the industry, you may know this as a national promotion, a corporate promotion, or a corporate discount.
node-oriented tree	A tree that is based on a detail structure, but the detail values are not used.
pagelet	Each block of content on the home page is called a pagelet. These pagelets display summary information within a small rectangular area on the page. The pagelet provide users with a snapshot of their most relevant PeopleSoft and non-PeopleSoft content.
participant	In PeopleSoft Enterprise Incentive Management, participants are recipients of the incentive compensation calculation process.
participant object	Each participant object may be related to one or more compensation objects. See also <i>compensation object</i> .
partner	A company that supplies products or services that are resold or purchased by the enterprise.
pay cycle	In PeopleSoft Payables, a set of rules that define the criteria by which it should select scheduled payments for payment creation.
pending item	In PeopleSoft Receivables, an individual receivable (such as an invoice, a credit memo, or a write-off) that has been entered in or created by the system, but hasn't been posted.

PeopleCode	PeopleCode is a proprietary language, executed by the PeopleSoft application processor. PeopleCode generates results based upon existing data or user actions. By using business interlink objects, external services are available to all PeopleSoft applications wherever PeopleCode can be executed.
PeopleCode event	An action that a user takes upon an object, usually a record field, that is referenced within a PeopleSoft page.
PeopleSoft Internet Architecture	The fundamental architecture on which PeopleSoft 8 applications are constructed, consisting of a relational database management system (RDBMS), an application server, a web server, and a browser.
performance measurement	In PeopleSoft Enterprise Incentive Management, a variable used to store data (similar to an aggregator, but without a predefined formula) within the scope of an incentive plan. Performance measures are associated with a plan calendar, territory, and participant. Performance measurements are used for quota calculation and reporting.
period context	In PeopleSoft Enterprise Incentive Management, because a participant typically uses the same compensation plan for multiple periods, the period context associates a plan context with a specific calendar period and fiscal year. The period context references the associated plan context, thus forming a chain. Each plan context has a corresponding set of period contexts.
plan	In PeopleSoft Sales Incentive Management, a collection of allocation rules, variables, steps, sections, and incentive rules that instruct the PeopleSoft Enterprise Incentive Management engine in how to process transactions.
plan context	In PeopleSoft Enterprise Incentive Management, correlates a participant with the compensation plan and node to which the participant is assigned, enabling the PeopleSoft Enterprise Incentive Management system to find anything that is associated with the node and that is required to perform compensation processing. Each participant, node, and plan combination represents a unique plan context—if three participants are on a compensation structure, each has a different plan context. Configuration plans are identified by plan contexts and are associated with the participants that refer to them.
plan template	In PeopleSoft Enterprise Incentive Management, the base from which a plan is created. A plan template contains common sections and variables that are inherited by all plans that are created from the template. A template may contain steps and sections that are not visible in the plan definition.
planned learning	In PeopleSoft Enterprise Learning Management, a self-service repository for all of a learner's planned learning activities and programs.
planning instance	In PeopleSoft Supply Planning, a set of data (business units, items, supplies, and demands) constituting the inputs and outputs of a supply plan.
portal registry	In PeopleSoft applications, the portal registry is a tree-like structure in which content references are organized, classified, and registered. It is a central repository that defines both the structure and content of a portal through a hierarchical, tree-like structure of folders useful for organizing and securing content references.
price list	In PeopleSoft Enterprise Pricer, enables you to select products and conditions for which the price list applies to a transaction. During a transaction, the system either determines the product price based on the predefined search hierarchy for the transaction or uses the product's lowest price on any associated, active price lists. This price is used as the basis for any further discounts and surcharges.
price rule	In PeopleSoft Enterprise Pricer, defines the conditions that must be met for adjustments to be applied to the base price. Multiple rules can apply when conditions of each rule are met.

price rule condition	In PeopleSoft Enterprise Pricer, selects the price-by fields, the values for the price-by fields, and the operator that determines how the price-by fields are related to the transaction.
price rule key	In PeopleSoft Enterprise Pricer, defines the fields that are available to define price rule conditions (which are used to match a transaction) on the price rule.
process category	In PeopleSoft Process Scheduler, processes that are grouped for server load balancing and prioritization.
process group	In PeopleSoft Financials, a group of application processes (performed in a defined order) that users can initiate in real time, directly from a transaction entry page.
process definition	Process definitions define each run request.
process instance	A unique number that identifies each process request. This value is automatically incremented and assigned to each requested process when the process is submitted to run.
process job	You can link process definitions into a job request and process each request serially or in parallel. You can also initiate subsequent processes based on the return code from each prior request.
process request	A single run request, such as a Structured Query Report (SQR), a COBOL or Application Engine program, or a Crystal report that you run through PeopleSoft Process Scheduler.
process run control	A PeopleTools variable used to retain PeopleSoft Process Scheduler values needed at runtime for all requests that reference a run control ID. Do not confuse these with application run controls, which may be defined with the same run control ID, but only contain information specific to a given application process request.
product category	In PeopleSoft Enterprise Incentive Management, indicates an application in the Enterprise Incentive Management suite of products. Each transaction in the PeopleSoft Enterprise Incentive Management system is associated with a product category.
programs	In PeopleSoft Enterprise Learning Management, a high-level grouping that guides the learner along a specific learning path through sections of catalog items. PeopleSoft Enterprise Learning Systems provides two types of programs—curricula and certifications.
progress log	In PeopleSoft Services Procurement, tracks deliverable-based projects. This is similar to the time sheet in function and process. The service provider contact uses the progress log to record and submit progress on deliverables. The progress can be logged by the activity that is performed, by the percentage of work that is completed, or by the completion of milestone activities that are defined for the project.
project transaction	In PeopleSoft Project Costing, an individual transaction line that represents a cost, time, budget, or other transaction row.
promotion	In PeopleSoft Promotions Management, a trade promotion, which is typically funded from trade dollars and used by consumer products manufacturers to increase sales volume.
publishing	In PeopleSoft Enterprise Incentive Management, a stage in processing that makes incentive-related results available to participants.
record group	A set of logically and functionally related control tables and views. Record groups help enable TableSet sharing, which eliminates redundant data entry. Record groups ensure that TableSet sharing is applied consistently across all related tables and views.
record input VAT flag	Abbreviation for <i>record input value-added tax flag</i> . Within PeopleSoft Purchasing, Payables, and General Ledger, this flag indicates that you are recording input VAT

on the transaction. This flag, in conjunction with the record output VAT flag, is used to determine the accounting entries created for a transaction and to determine how a transaction is reported on the VAT return. For all cases within Purchasing and Payables where VAT information is tracked on a transaction, this flag is set to Yes. This flag is not used in PeopleSoft Order Management, Billing, or Receivables, where it is assumed that you are always recording only output VAT, or in PeopleSoft Expenses, where it is assumed that you are always recording only input VAT.

record output VAT flag	Abbreviation for <i>record output value-added tax flag</i> . See <i>record input VAT flag</i> .
reference data	In PeopleSoft Sales Incentive Management, system objects that represent the sales organization, such as territories, participants, products, customers, channels, and so on.
reference object	In PeopleSoft Enterprise Incentive Management, this dimension-type object further defines the business. Reference objects can have their own hierarchy (for example, product tree, customer tree, industry tree, and geography tree).
reference transaction	In commitment control, a reference transaction is a source transaction that is referenced by a higher-level (and usually later) source transaction, in order to automatically reverse all or part of the referenced transaction's budget-checked amount. This avoids duplicate postings during the sequential entry of the transaction at different commitment levels. For example, the amount of an encumbrance transaction (such as a purchase order) will, when checked and recorded against a budget, cause the system to concurrently reference and relieve all or part of the amount of a corresponding pre-encumbrance transaction, such as a purchase requisition.
regional sourcing	In PeopleSoft Purchasing, provides the infrastructure to maintain, display, and select an appropriate vendor and vendor pricing structure that is based on a regional sourcing model where the multiple ship to locations are grouped. Sourcing may occur at a level higher than the ship to location.
relationship object	In PeopleSoft Enterprise Incentive Management, these objects further define a compensation structure to resolve transactions by establishing associations between compensation objects and business objects.
remote data source data	Data that is extracted from a separate database and migrated into the local database.
REN server	Abbreviation for <i>real-time event notification server</i> in PeopleSoft MultiChannel Framework.
requester	In PeopleSoft eSettlements, an individual who requests goods or services and whose ID appears on the various procurement pages that reference purchase orders.
role	Describes how people fit into PeopleSoft Workflow. A role is a class of users who perform the same type of work, such as clerks or managers. Your business rules typically specify what user role needs to do an activity.
role user	A PeopleSoft Workflow user. A person's role user ID serves much the same purpose as a user ID does in other parts of the system. PeopleSoft Workflow uses role user IDs to determine how to route worklist items to users (through an email address, for example) and to track the roles that users play in the workflow. Role users do not need PeopleSoft user IDs.
roll up	In a tree, to roll up is to total sums based on the information hierarchy.
run control	A run control is a type of online page that is used to begin a process, such as the batch processing of a payroll run. Run control pages generally start a program that manipulates data.
run control ID	A unique ID to associate each user with his or her own run control table entries.

run-level context	In PeopleSoft Enterprise Incentive Management, associates a particular run (and batch ID) with a period context and plan context. Every plan context that participates in a run has a separate run-level context. Because a run cannot span periods, only one run-level context is associated with each plan context.
search query	You use this set of objects to pass a query string and operators to the search engine. The search index returns a set of matching results with keys to the source documents.
section	In PeopleSoft Enterprise Incentive Management, a collection of incentive rules that operate on transactions of a specific type. Sections enable plans to be segmented to process logical events in different sections.
security event	In commitment control, security events trigger security authorization checking, such as budget entries, transfers, and adjustments; exception overrides and notifications; and inquiries.
serial genealogy	In PeopleSoft Manufacturing, the ability to track the composition of a specific, serial-controlled item.
serial in production	In PeopleSoft Manufacturing, enables the tracing of serial information for manufactured items. This is maintained in the Item Master record.
session	In PeopleSoft Enterprise Learning Management, a single meeting day of an activity (that is, the period of time between start and finish times within a day). The session stores the specific date, location, meeting time, and instructor. Sessions are used for scheduled training.
session template	In PeopleSoft Enterprise Learning Management, enables you to set up common activity characteristics that may be reused while scheduling a PeopleSoft Enterprise Learning Management activity—characteristics such as days of the week, start and end times, facility and room assignments, instructors, and equipment. A session pattern template can be attached to an activity that is being scheduled. Attaching a template to an activity causes all of the default template information to populate the activity session pattern.
setup relationship	In PeopleSoft Enterprise Incentive Management, a relationship object type that associates a configuration plan with any structure node.
share driver expression	In PeopleSoft Business Planning, a named planning method similar to a driver expression, but which you can set up globally for shared use within a single planning application or to be shared between multiple planning applications through PeopleSoft Enterprise Warehouse.
single signon	With single signon, users can, after being authenticated by a PeopleSoft application server, access a second PeopleSoft application server without entering a user ID or password.
source transaction	In commitment control, any transaction generated in a PeopleSoft or third-party application that is integrated with commitment control and which can be checked against commitment control budgets. For example, a pre-encumbrance, encumbrance, expenditure, recognized revenue, or collected revenue transaction.
SpeedChart	A user-defined shorthand key that designates several ChartKeys to be used for voucher entry. Percentages can optionally be related to each ChartKey in a SpeedChart definition.
SpeedType	A code representing a combination of ChartField values. SpeedTypes simplify the entry of ChartFields commonly used together.
staging	A method of consolidating selected partner offerings with the offerings from the enterprise's other partners.

statutory account	Account required by a regulatory authority for recording and reporting financial results. In PeopleSoft, this is equivalent to the Alternate Account (ALTACCT) ChartField.
step	In PeopleSoft Sales Incentive Management, a collection of sections in a plan. Each step corresponds to a step in the job run.
storage level	In PeopleSoft Inventory, identifies the level of a material storage location. Material storage locations are made up of a business unit, a storage area, and a storage level. You can set up to four storage levels.
subcustomer qualifier	A value that groups customers into a division for which you can generate detailed history, aging, events, and profiles.
Summary ChartField	You use summary ChartFields to create summary ledgers that roll up detail amounts based on specific detail values or on selected tree nodes. When detail values are summarized using tree nodes, summary ChartFields must be used in the summary ledger data record to accommodate the maximum length of a node name (20 characters).
summary ledger	An accounting feature used primarily in allocations, inquiries, and PS/nVision reporting to store combined account balances from detail ledgers. Summary ledgers increase speed and efficiency of reporting by eliminating the need to summarize detail ledger balances each time a report is requested. Instead, detail balances are summarized in a background process according to user-specified criteria and stored on summary ledgers. The summary ledgers are then accessed directly for reporting.
summary time period	In PeopleSoft Business Planning, any time period (other than a base time period) that is an aggregate of other time periods, including other summary time periods and base time periods, such as quarter and year total.
summary tree	A tree used to roll up accounts for each type of report in summary ledgers. Summary trees enable you to define trees on trees. In a summary tree, the detail values are really nodes on a detail tree or another summary tree (known as the <i>basis</i> tree). A summary tree structure specifies the details on which the summary trees are to be built.
syndicate	To distribute a production version of the enterprise catalog to partners.
system function	In PeopleSoft Receivables, an activity that defines how the system generates accounting entries for the general ledger.
TableSet	A means of sharing similar sets of values in control tables, where the actual data values are different but the structure of the tables is the same.
TableSet sharing	Shared data that is stored in many tables that are based on the same TableSets. Tables that use TableSet sharing contain the SETID field as an additional key or unique identifier.
target currency	The value of the entry currency or currencies converted to a single currency for budget viewing and inquiry purposes.
template	A template is HTML code associated with a web page. It defines the layout of the page and also where to get HTML for each part of the page. In PeopleSoft, you use templates to build a page by combining HTML from a number of sources. For a PeopleSoft portal, all templates must be registered in the portal registry, and each content reference must be assigned a template.
territory	In PeopleSoft Sales Incentive Management, hierarchical relationships of business objects, including regions, products, customers, industries, and participants.
TimeSpan	A relative period, such as year-to-date or current period, that can be used in various PeopleSoft General Ledger functions and reports when a rolling time frame, rather

	than a specific date, is required. TimeSpans can also be used with flexible formulas in PeopleSoft Projects.
trace usage	In PeopleSoft Manufacturing, enables the control of which components will be traced during the manufacturing process. Serial- and lot-controlled components can be traced. This is maintained in the Item Master record.
transaction allocation	In PeopleSoft Enterprise Incentive Management, the process of identifying the owner of a transaction. When a raw transaction from a batch is allocated to a plan context, the transaction is duplicated in the PeopleSoft Enterprise Incentive Management transaction tables.
transaction state	In PeopleSoft Enterprise Incentive Management, a value assigned by an incentive rule to a transaction. Transaction states enable sections to process only transactions that are at a specific stage in system processing. After being successfully processed, transactions may be promoted to the next transaction state and “picked up” by a different section for further processing.
Translate table	A system edit table that stores codes and translate values for the miscellaneous fields in the database that do not warrant individual edit tables of their own.
tree	The graphical hierarchy in PeopleSoft systems that displays the relationship between all accounting units (for example, corporate divisions, projects, reporting groups, account numbers) and determines roll-up hierarchies.
unclaimed transaction	In PeopleSoft Enterprise Incentive Management, a transaction that is not claimed by a node or participant after the allocation process has completed, usually due to missing or incomplete data. Unclaimed transactions may be manually assigned to the appropriate node or participant by a compensation administrator.
universal navigation header	Every PeopleSoft portal includes the universal navigation header, intended to appear at the top of every page as long as the user is signed on to the portal. In addition to providing access to the standard navigation buttons (like Home, Favorites, and signoff) the universal navigation header can also display a welcome message for each user.
user interaction object	In PeopleSoft Sales Incentive Management, used to define the reporting components and reports that a participant can access in his or her context. All Sales Incentive Management user interface objects and reports are registered as user interaction objects. User interaction objects can be linked to a compensation structure node through a compensation relationship object (individually or as groups).
variable	In PeopleSoft Sales Incentive Management, the intermediate results of calculations. Variables hold the calculation results and are then inputs to other calculations. Variables can be plan variables that persist beyond the run of an engine or local variables that exist only during the processing of a section.
VAT exception	Abbreviation for <i>value-added tax exception</i> . A temporary or permanent exemption from paying VAT that is granted to an organization. This terms refers to both VAT exoneration and VAT suspension.
VAT exempt	Abbreviation for <i>value-added tax exempt</i> . Describes goods and services that are not subject to VAT. Organizations that supply exempt goods or services are unable to recover the related input VAT. This is also referred to as exempt without recovery.
VAT exoneration	Abbreviation for <i>value-added tax exoneration</i> . An organization that has been granted a permanent exemption from paying VAT due to the nature of that organization.
VAT suspension	Abbreviation for <i>value-added tax suspension</i> . An organization that has been granted a temporary exemption from paying VAT.
warehouse	A PeopleSoft data warehouse that consists of predefined ETL maps, data warehouse tools, and DataMart definitions.

work order	In PeopleSoft Services Procurement, enables an enterprise to create resource-based and deliverable-based transactions that specify the basic terms and conditions for hiring a specific service provider. When a service provider is hired, the service provider logs time or progress against the work order.
worksheet	A way of presenting data through a PeopleSoft Business Analysis Modeler interface that enables users to do in-depth analysis using pivoting tables, charts, notes, and history information.
worklist	The automated to-do list that PeopleSoft Workflow creates. From the worklist, you can directly access the pages you need to perform the next action, and then return to the worklist for another item.
XML schema	An XML definition that standardizes the representation of application messages, component interfaces, or business interlinks.
yield by operation	In PeopleSoft Manufacturing, the ability to plan the loss of a manufactured item on an operation-by-operation basis.
zero-rated VAT	Abbreviation for <i>zero-rated value-added tax</i> . A VAT transaction with a VAT code that has a tax percent of zero. Used to track taxable VAT activity where no actual VAT amount is charged. Organizations that supply zero-rated goods and services can still recover the related input VAT. This is also referred to as exempt with recovery.

Index

A

- A4 paper, printing 107
- additional documentation xiv
- Adobe Reader 122
- Agfa Monotype 52
- ALTER-PRINTER command 117, 118
- Alternate Character Architecture 59
- alternate language DLLs
 - delivered 209, 212
 - locating language resources 211
 - translating resource files 212
 - understanding 210
- American National Standards Institute (ANSI), *See* ANSI character sets
- ANSI character sets
 - forcing binary sorts in SQL 97
 - non-Unicode vs Unicode data, identifying 81
 - understanding 42, 45
- Append_Report_Translation function 110
- Application Data page 245
- Application Designer
 - defining the text search scope 184
 - setting fonts 53
 - translating definitions in projects 218
 - translating folder tabs 229
 - Translation Designer 217
 - See Also* Translation Designer
 - understanding field length semantics 55
- Application Engine definitions, translating 243
- application fundamentals xiii
- application servers
 - locating language resources 211
 - selecting character sets 51
 - supporting email character sets 54
 - using MAKEUNIX.BAT 214
- Arabic
 - character set support 126
 - SQR support 121
 - user interface support 49
- ASCII
 - native2ascii utility 207
 - saving properties files 207

- sorting data 93
- understanding 36
- UTF-8 encoding 40
- auditing report, base language swap 159

B

- Baltic character set 126
- base languages
 - designing base-language pages 144
 - setting in Configuration Manager 17
 - swapping 158, 161
 - swapping, checking space/resources for 161
 - swapping, prerequisites for 159
 - swapping, running the audit report for 159
 - understanding 149
 - understanding base language tables 149
- base records
 - fixing invalid related language records 159
 - fixing key fields 160
- base tables
 - cloning 152
 - fixing orphaned rows 160
 - maintaining 169
 - related language table and base table – one-to-one view 164
 - related language table and base tables – one-to-two view 165
 - related language tables and base tables – two-to-two view 166
 - related language tables, associating with 153
 - related language tables, understanding 152
 - storing language-sensitive descriptions 153
 - understanding 150
- base time zones
 - %BaseTimeZone system variable 33
 - changing 28
 - setting 26
 - understanding 25
 - %BaseTimeZone system variable 32, 33

Basic Multilingual Plane (BMP) 39
 bi-directional text
 Arabic character set support 126
 Hebrew character set support 129
 printing in SQR for PeopleSoft 121
 SQR support 121
 user interface support 49
 binary sorting
 forcing binary sorts in SQL 97
 understanding 93
 using 96
 %BINARYSORT meta-SQL
 binary sorting 96
 forcing binary sorts in SQL 97
 sorting in COBOL 69, 98
 BIND/SELECT log 79
 Bitstream 52
 BMP 39
 browsers
 Internet Explorer 52
 See Also Internet Explorer
 managing character sets/fonts 52
 Netscape Navigator 53
 See Also Netscape Navigator
 setting language preferences 15
 setting up fonts in SQR for
 PeopleSoft 114
 business interlinks, translating 232

C

cbl2uni.bat command 71
 Celtic character set 127
 channels, translating 232
 character sets
 ANSI 42
 See Also ANSI character sets
 Arabic 126
 See Also Arabic
 Baltic 126
 Celtic 127
 Chinese 127
 See Also Chinese
 COBOL, encodings for 67
 COBOL, supporting for 44
 Cyrillic 128
 databases, selecting for 50
 databases, supporting in 44
 DBCS 42
 See Also DBCS
 Greek 128

Gurmukhi 129
 Hebrew 129
 See Also Hebrew
 Icelandic 130
 Japanese 59
 See Also Japanese
 Korean 131
 See Also Korean
 Latin 131
 Malayalam 134
 managing in PeopleSoft Pure Internet
 Architecture 43, 52
 non-Unicode databases, checking field
 length for 57
 non-Unicode, converting from Unicode
 to 44
 non-Unicode, determining
 appropriate 201
 non-Unicode, understanding 42
 See Also ANSI character sets
 Nordic 134
 performing file operations 44
 SBCS 42
 See Also SBCS
 selecting for application servers 51
 selecting for CSV queries 205
 selecting for SQR messages 113
 selecting for the Verity search
 engine 205
 selecting Linux default 51
 selecting UNIX default 51
 selecting Windows default 51
 setting default 205
 setting for files being opened 124
 setting up files for PostScript
 printing 120
 setting/supporting for email 54
 Shift-JIS 39
 See Also Shift-JIS
 Slavic 135
 supporting for third-party products 44
 supporting in SQR for PeopleSoft 124
 Symbol 135
 Thai 135
 Turkish 135
 understanding 35, 36
 understanding encodings 125
 understanding field length
 semantics 55
 understanding keyboard layouts 53

- understanding types 45
- Unicode 38
 - See Also* Unicode
 - Vietnamese 136
- character strings
 - comparing in COBOL 98
 - detecting/convertng characters 59
 - identifying ANSI vs Unicode data 81
 - measuring length 57, 123
 - parsing Unicode 69, 83
 - requesting maps for 86
 - shifting DBCSs for character sets 43
- characters
 - character sets 35
 - See Also* character sets
 - sorting 93
 - See Also* sorting
 - strings 123
 - See Also* character strings
 - using digit group separators 23
- chcp command 51
- Chinese
 - browser limitations for data display 53
 - printing in PDF format 116, 117
 - printing with PostScript fonts 120
 - setting up fonts 115
 - sorting 94
 - standard name formats 58
 - supporting character set encodings 127
 - transforming keyboard strokes into ideographs 54
- %clienttimezone system variable 33
- %ClientTimeZone system variable 32
- COBOL
 - binary sorting 96
 - converting programs to run on Unicode 70
 - See Also* COBOL Conversion utility
 - identifying ANSI vs Unicode data 81
 - parsing Unicode strings 69, 83
 - PTPSTRFN function 83
 - See Also* PTPSTRFN function
 - setting column lengths in dynamic SQL 81
 - sorting 69, 98
 - storage requirements for Unicode 68
 - supporting character sets 44
 - Unicode encodings in 67
 - Unicode-specific error messages 69
- COBOL Conversion utility
 - expanding fields 72
 - fine-tuning programs 80
 - handling arrays comprising a single character 76, 83
 - handling date/time fields 75
 - handling redefined character fields 75
 - handling SQL buffer setup data 72, 81
 - identifying converted programs 72
 - running 71
 - understanding 70
 - understanding error logs 79
 - using directives 77
- COCCUR directive 77, 78
- codepages, *See* character sets
- codepoints 35
- Cognos PowerPlay, *See* PowerPlay
- collation, *See* sorting
- comma-separated value (CSV) format, *See* CSV format
- commands
 - ALTER-PRINTER 117, 118
 - cbl2uni.bat 71
 - chcp 51
 - COBOL Conversion utility 71
 - DEFINE-PRINTER 118
 - MAKEALTL 214
 - pscbl.mak 71
 - SQR Open 124
 - SQR OPEN 113, 124
 - SWAP_BASE_LANGUAGE 158, 161
- comments, submitting xvii
- common elements xvii
- component interfaces messages, translating 231
- Configuration Manager
 - adding languages 205
 - setting fonts 53
 - setting languages 17
- contact information xvii
- ConvertDatetimeToBase function 33
- ConvertTimeToBase function 33
- CP1250 120
- cross-references xvi
- CSV format
 - downloading query results 204
 - report output 113, 121
 - selecting character sets for queries 205
 - understanding 125
- Cube Manager
 - understanding cubes 102

- using Essbase cubes 103
- using PowerPlay cubes 102
- currencies
 - currency fields 61
 - See Also* currency fields
 - setting precision 108
 - setting the display for PeopleTools 23
- Currency Code table 61
- currency fields
 - connecting to the system-wide multicurrency check box 64
 - enabling system-wide settings 64
 - hiding 63
 - setting up 62
 - setting up display properties 62
 - sizing 65
 - understanding formatting 61
 - understanding multicurrency settings 63
- %CurrentDateTime meta-SQL 25
- Customer Connection website xiv
- Cyrillic character set 128

D

- Data Mover
 - adding translations to databases 156
 - listing translatable tables 185
 - making tables searchable 197
 - swapping base languages 158, 161
- databases
 - adding translations 156
 - checking field length for non-Unicode 57
 - selecting character sets 50
 - supporting character sets 44
 - swapping base languages 158, 159
 - understanding field length semantics 55
 - upgrading customizations to multilingual 157
- dates
 - datetime field time zones 31
 - exception date log 80
 - formatting in SQRs 108
 - handling date/datetime fields in the COBOL Conversion utility 75
- DATETIME.SQC program 108
- Daylight Savings page 30
- daylight savings time
 - setting for time zone definitions 29

- understanding 25
- viewing start/end dates 30
- DB2 UDB for Linux, Unix, and Windows (Unicode support) 40
- DB2 UDB for OS/390 and z/OS
 - creating related language tables 162
 - shifting DBCSs for character sets 43
 - sorting in COBOL 69
 - understanding field length semantics 56
- DBCS
 - checking field length for non-Unicode databases 57
 - checking field lengths on Japanese databases 57
 - understanding 42
- decimal precision
 - setting for currencies 61, 108
 - understanding multicurrency settings 63
 - using the International Field Size utility 65
- Define Personalizations page 23
- DEFINE-PRINTER command 118
- definitions
 - accessing 223
 - cloning search 187
 - finding references 223
 - language 199
 - opening 223
 - string 105
 - See Also* string definitions
 - translating 217
 - translating Application Engine 243
 - translating business process 231
 - translating HTML 247
 - translating in order 224
 - translating menu 227
 - translating message 243
 - translating page 225
 - See Also* page definitions
 - translating Process Scheduler 232
 - translating record 240
 - translating tree 238
 - using translation pages 224
 - viewing in Translation Designer 219
 - viewing time zone 28
- documentation
 - printed xiv
 - related xiv

updates xiv
 double-byte character sets (DBCS), *See*
 DBCS

E

email character sets 39, 54
 environment variables
 INCLUDE 213
 running MAKEALTL.BAT 213
 running MAKEUNIX.BAT 214
 TOOLBIN 213
 errors
 adding error property files for
 languages 206
 BIND/SELECT log 79
 COBOL conversion logs 79
 decimal precision 63
 exception date log 80
 exception log 79
 running COBOL Unicode 69
 running the audit report for base language
 swaps 159
 swapping base languages 161
 Essbase
 Unicode support 44
 using multiple-language cubes 103
 EUDCEDIT program 116
 Excel
 copying translations from Translation
 Designer 223
 international versions 141
 SQR for PeopleSoft output 125
 translating PS/nVision layout to 137
 exception date log 80
 exception log 79
 EXPAND directive 77
 EXPEOF directive 77, 78

F

fields
 currency 61
 See Also currency fields
 numeric 20
 See Also numeric fields
 page 32
 See Also page fields
 record 144
 See Also record fields
 translating definitions 219

translating labels 143
 using the Get_Field_Information
 procedure 110

flat files 125

folder tabs, translating 229

fonts

 foundries 52

 managing in PeopleSoft Pure Internet
 Architecture 52

 obtaining 52

 PDF 116

See Also PDF fonts

 printing limitations in SQR for
 PeopleSoft 121

 public domain 52

 setting in Configuration Manager 53

 setting in PeopleCode Editor 53

 setting up files for PostScript
 printing 119

 setting up in SQR for PeopleSoft 114

Format_Currency_Amt function 108

Format_Currency_Amt_Numeric
 function 108

functions

 Append_Report_Translation 110

 ConvertDatetimeToBase 33

 ConvertTimeToBase 33

 Format_Currency_Amt 108

 Format_Currency_Amt_Numeric 108

 Get_Field_Information 110

 Init_Report_Translation 109

 IsDaylightSavings 33

 Length() 123

 Lengthp() 123

 Lengtht() 124

 SetLanguage 13

 string 59

 Substr() 123

 Substrp() 123

 Substrt() 124

G

General Profile Information page 14

Get_Field_Information function 110

Global Multi-Language CD-ROM 155

Global Technology

 implementing 4

 understanding xix, 3

globalization

 designing global-ready pages 144

- developing global applications 143
- displaying global currency fields 63
- Global Technology xix
 - See Also* Global Technology
- PS/nVision design-time globalization features 137
- SQR for PeopleSoft 107
 - See Also* SQR for PeopleSoft
- glossary 269
- GMT 25
- Greek character set 128
- Greenwich mean time (GMT) 25
- Gurmukhi character set 129

H

- Hebrew
 - character set support 129
 - SQR support 121
 - user interface support 49
- Help documentation 15
- HTML
 - pages 52
 - See Also* HTML pages
 - report output 113, 121, 126
 - translating definitions 247
- HTML pages
 - character set support 40
 - managing character sets/fonts 52
- Hyperion Essbase, *See* Essbase

I

- Icelandic character set 130
- images
 - adding to languages 207
 - translating 216
- IME 52, 53
- INCLUDE environment variable 213
- %include% environment variable 214
- Init_Report_Translation function 109
- input method editor (IME) 52, 53
- Inquiry pages 195
- international
 - handling international characters 58
 - preferences 11
 - text 112
 - See Also* international text
- International Field Size utility 65
- International Field Sizes table 65

- International Organization for Standardization (ISO), *See* ISO
- International Preferences page 13
- international text
 - printing in SQRs 113
 - processing 112
 - setting up files for PCL/line printing 118
 - setting up files for PostScript printing 119
 - using in SQR for PeopleSoft 121, 122

- Internet Explorer
 - downloading/matching fonts 52
 - preferred languages 15
 - viewing Japanese/Chinese/Korean data 53

- IsDaylightSavings function 33

ISO

- character sets 35
- locales 15
 - See Also* ISO locales

ISO locales

- determining language codes 200
- selecting for the PSLANGUAGES table 205
- understanding language codes 15

J**Japanese**

- browser limitations for data display 53
- character sets for outgoing email 54
- checking field lengths 57
- default non-Unicode character set 51
- detecting/converting characters 59
- formatting report dates/times 108
- non-Unicode DBCS settings 58
- printing in PDF format 116, 117
- printing with PostScript fonts 120
- setting up fonts 115
- setting up PCL printing 118
- setting up user-defined fonts 116
- Shift-JIS character set 39
 - See Also* Shift-JIS
- sorting 94
- standard name formats 58
- supporting character set encodings 130
- transforming keyboard strokes into ideographs 54

Java

- native2ascii utility 207

See Also native2ascii utility
 translating properties files 215

K

key fields
 designing related language records 162
 fixing errors for base language
 swaps 159
 keyboard layouts, foreign language 52,
 53
 Korean
 browser limitations for data display 53
 printing in PDF format 116, 117
 printing with PostScript fonts 120
 setting up fonts 115
 sorting 94
 standard name formats 58
 supporting character set encodings 131
 transforming keyboard strokes into
 ideographs 54

L

labels
 designing for global-ready pages 144
 labelling time zones 28
 retrieving in PS/nVision 140
 translating 143
 translating for mobile page views 234
 translating PS/nVision layout to
 Excel 137
 using the Strings table 104
 See Also Strings table
 %Language system variable 13
 %Language_Base system variable 13
 LANGUAGE_CD field 202
 languages
 adding error/text property files 206
 adding for signon 206
 adding images 207
 adding to Configuration Manager 205
 adding to the PSLANGUAGES
 table 203
 base 144
 See Also base languages
 choosing for searches 183
 codes, adding 199, 202
 codes, determining 200
 codes, understanding PeopleSoft
 /ISO 15

cubes 102
 See Also Cube Manager
 databases, using non-Unicode 50
 defining locales 21
 definitions 199
 enabling 205
 fonts 52
 See Also fonts
 Global Multi-Language CD-ROM 155
 language-specific configuration
 files 112
 Multi Language Entry 171
 See Also Multi Language Entry
 PeopleTools Language Development
 Kit 209
 preferences for browsers, applying 15
 preferences, setting/changing 12
 preferences, setting/changing
 Windows 17
 preferences, understanding 11
 profiles, controlling user 14
 profiles, overriding user 14
 PS/nVision, setting for 137
 queries, language-sensitive 101
 queries, scheduled 102
 related language records 159
 See Also related language records
 related language tables 149
 See Also related language tables
 selecting for spell check 205
 session languages, changing 13
 session languages, choosing 12
 session languages, understanding 12
 setting in Configuration Manager 17
 setting up files for PCL/line
 printing 118
 setting up files for PostScript
 printing 119
 setting up PDF fonts for SQRs 116
 sorting 93
 See Also sorting
 Terminology Management tool 177
 See Also Terminology Management
 tool
 translating 143
 See Also translation
 using the Append_Report_Translation
 procedure 110
 using the Get_Field_Information
 procedure 110

- using the Init_Report_Translation procedure 109
- using the Strings table 104

- See Also* Strings table

- Latin character set 131

- Latin2 120

- Length() function 123

- Lengthp() function 123

- Lengtht() function 124

- %lib% environment variable 214

- licensing

- fonts 52

- IMEs 54

- Linux

- compiling COBOL 71

- determining character sets 201

- locating language resources 211

- PSSQR.UNX 112

- See Also* PSSQR files

- selecting default character sets 51

- setting up files for PostScript printing 120

- using MAKEUNIX.BAT 214

- Locale Defaults page 22

- Locale Definition page 21

- locales

- defining 21

- ISO 15

- See Also* ISO locales

- setting locale-based formatting 22

- understanding locale-based

- formatting 18

- Verity 205

M

- MAKEALTL command 214

- MAKEALTL.BAT 213

- MAKEUNIX.BAT 214

- Malayalam character set 134

- Manage Installed Languages page 203

- menus

- translating definitions 227

- translating items 228

- messaging

- replacing message text 191

- translating channels 232

- translating component interface messages 231

- translating message definitions 243

- translating messages 226

- meta-SQL

- %BINARYSORT 69

- See Also* %BINARYSORT meta-SQL

- %CurrentDateTime 25

- Microsoft Excel, *See* Excel

- Microsoft Internet Explorer, *See* Internet Explorer

- Microsoft Windows, *See* Windows

- MMA Partners xiv

- mobile pages

- translating data 234

- translating items 235

- translating page names/view labels 234

- MSSQL Server

- SQL ORDER BY sorts 95

- supporting Unicode 40

- understanding field length

- semantics 56

- Multi Language Entry

- editing data 170

- enabling 171

- entering data 172

- understanding 171

- My Personalizations page

- defining user-specific formatting 20

- understanding locale-based

- formatting 19

N

- native2ascii utility 207

- Netscape Navigator

- downloading/matching fonts 53

- viewing Japanese/Chinese/Korean data 53

- NOCBGN directive 77

- NOCEND directive 77

- NOCLN directive 77

- nonshifting DBCSs 42

- Nordic character set 134

- notes xvi

- numeric fields

- defining user-specific formatting 20

- handling in the COBOL Conversion utility 75

- setting system-wide formatting 23

O

- objects

- adding translations 156

- designing for easy translation 143
 - modifying searchable 197
 - selecting search objects 196
 - storing in multiple languages 149
 - translating 215
 - translating Application Engine 243
 - translating business interlink 232
 - translating business process 230
 - translating menu 227
 - translating message 226, 243
 - translating mobile 234
 - translating portal 237
 - translating Process Scheduler 232
 - translating query 241
 - translating record 240
 - translating string 242
 - upgrade issues 197
 - viewing searchable 196
- Oracle
- checking space/resources for base language swaps 161
 - SQL ORDER BY sorts 95
 - supporting Unicode 40
 - understanding field length semantics 55
- P**
- page definitions
 - translating 219, 225
 - understanding 217
 - Page Field Properties dialog box 33
 - page fields
 - checking field length for non-Unicode databases 57
 - enabling length checking 57
 - setting time zones 32, 33
 - understanding field length semantics 55
 - pages
 - designing global-ready 144
 - entering data in multiple languages 172
 - fields 32
 - See Also* page fields
 - freezing layout 144
 - HTML 52
 - See Also* HTML pages
 - search status 180
 - sizing/ordering controls 144
 - translating labels 143
 - translating review page text 244
 - using translation pages 224
 - %path% environment variable 214
 - PCL printing 118, 125
 - PDF fonts
 - printing Chinese, Japanese, and Korean text 117
 - setting up for SQRs 116
 - setting up Japanese user-defined fonts 116
 - PeopleBooks
 - ordering xiv
 - PeopleCode
 - binary sorting 96
 - forcing binary sorts in SQL 97
 - functions 13
 - See Also* functions
 - maintaining programs on related language records 163
 - meta-SQL 96
 - See Also* meta-SQL
 - setting fonts 53
 - PeopleCode, typographical conventions xv
 - PeopleSoft Application Designer, *See* Application Designer
 - PeopleSoft application fundamentals xiii
 - PeopleSoft COBOL, *See* COBOL
 - PeopleSoft Configuration Manager, *See* Configuration Manager
 - PeopleSoft Cube Manager, *See* Cube Manager
 - PeopleSoft Data Mover, *See* Data Mover
 - PeopleSoft nVision Layout Definition dialog box 138
 - PeopleSoft Pure Internet Architecture
 - managing character sets 43, 52
 - managing fonts 52
 - supporting IMEs 54
 - PeopleSoft Query 101
 - PeopleTools
 - Global Technology xix
 - See Also* Global Technology
 - handling international characters 58
 - installing PeopleSoft-provided translations 155
 - Language Development Kit 209
 - setting options 26
 - See Also* PeopleTools Options page
 - sorting 93
 - See Also* sorting

- supporting Unicode 39
 - Terminology Management tool 177
 - See Also* Terminology Management tool
 - translating 209
 - See Also* translation
 - PeopleTools Options page
 - enabling field length checking 57
 - enabling system-wide currency settings 64
 - Multi-Currency check box 63, 64
 - setting base time zones 26
 - setting sort orders 97
 - personalization, *See* My Personalizations page
 - Plane 0 39
 - portals, translating 237
 - PostScript printing 119, 125
 - PowerPlay
 - Unicode support 44
 - using local-language cubes 102
 - prerequisites xiii
 - printed documentation xiv
 - printing
 - international text in SQR for PeopleSoft 121
 - international text in SQRs 113
 - setting up files for PCL/line 118
 - setting up files for PostScript printing 119
 - using A4 paper 107
 - Process Scheduler
 - language-specific configuration files 112
 - translating definitions 232
 - programs
 - DATETIME.SQC 108
 - EUDCEDIT 116
 - SETENV.SQC 108
 - SQRTRANS.SQC 109
 - projects
 - translating definitions in 218
 - upgrading customizations to multilingual databases 157
 - viewing in Translation Designer 219
 - properties files
 - adding for languages 206
 - saving/formatting 207
 - translating 215
 - using the native2ascii utility 207
 - PS/nVision
 - design-time globalization features 137
 - retrieving labels 140
 - run-time language features 139
 - searching for files 139
 - setting up string criteria 138
 - translating reports 139
 - understanding 137
 - variables 140
 - See Also* PS/nVision variables
 - PS/nVision Report Request page 139
 - PS/nVision variables
 - evaluating 105
 - understanding 140
 - pscbl.mak command 71
 - PSLANGUAGES table
 - adding new languages 203
 - determining the signon language 15
 - PSPNLFIELD table 161
 - PSSQR files
 - advanced settings 113
 - checking encoding settings 122
 - setting up files for PostScript printing 119
 - setting up fonts 114
 - setting up PDF fonts 116
 - understanding 112
 - PSTZOFFSET table 29
 - %PSVER% environment variable 214
 - PTPDYSQL program 81
 - PTPSTRFN function
 - requesting character string map 86
 - requesting single character length 83
 - understanding 83
- Q**
- queries
 - generating time zone offsets 29
 - language-sensitive 101
 - languages for scheduled 102
 - translating 241
- R**
- Record Field Properties dialog box 32
 - record fields
 - associating with group boxes 144
 - setting time zones 32
 - records
 - base 160

- See Also* base records
- record fields 144
 - See Also* record fields
- related language 159
 - See Also* related language records
 - translating 240
- Regional Options dialog box 23
- related documentation xiv
- related language records
 - designing 162
 - fixing invalid 159
 - making LANGUAGE_CD a key field 159
- related language tables
 - associating with base tables 153
 - base and related language – one-to-one view 164
 - base tables and related language table – two-to-one view 165
 - base tables and related language tables – two-to-two view 166
 - cloning 152
 - creating 162
 - fixing swap errors 160
 - maintaining 169
 - requirements 152
 - storing translations 153
 - structure 152
 - understanding 149
 - using 154
- related language views
 - creating 163
 - fixing wrong key structures 159
 - one base table and one related language table 164
 - two base tables and one related language table 165
 - two base tables and two related language tables 166
- related languages
 - records 159
 - See Also* related language records
 - tables 149
 - See Also* related language tables
 - views 159
 - See Also* related language views
- Replace Data page 189
- Replace Messages page 191
- Replace process
 - disabling 184, 191

- enabling 191
- replacing message text 191
- replacing non-message text 189
- reviewing replacement text 193
- running 192
- understanding 179
- undoing replacements 193
- verifying replacement text 190
- viewing/replacing search results 188
- Replace Results/Undo page 193, 195
- reports
 - PS/nVision 137
 - See Also* PS/nVision
 - SQR for PeopleSoft 107
 - See Also* SQR for PeopleSoft

S

- SBCS
 - checking field length for non-Unicode databases 57
 - understanding 42
- Search Profile page 182, 195
- Search Results/Replace page 192
- searches
 - choosing languages 183
 - criteria, cloning 187
 - criteria, defining 182
 - criteria, reviewing 195
 - criteria, sample 187
 - defining text 185
 - defining the scope 184
 - enabling processing 183
 - granting search privileges 181
 - objects, modifying searchable 197
 - objects, selecting 196
 - objects, viewing searchable 196
 - page search status 180
 - results, reviewing 189
 - results, viewing all 195
 - results, viewing search-only 195
 - results, viewing/replacing 188
 - running searches 188
 - searching/replacing in Translation Designer 222
 - Terminology Management tool 178
- security
 - accessing definitions 223
 - granting search privileges 181
 - personalizing locale-based formatting 19

- Set International Field Sizes SQR
 - utility 65
- SETENV.SQC program 108
- SetLanguage function 13
- Shift-JIS
 - checking field length for non-Unicode databases 57
 - forcing binary sorts in SQL 97
 - limitations 39
 - nonshifting DBCSs 43
 - understanding 38
- shifting DBCSs 43
- signon
 - adding languages for 206
 - applying browser language preferences for 15
 - overriding user profile languages 14
 - session languages, choosing 12
 - session languages, understanding 12
- Simple Mail Transfer Protocol (SMTP)
 - character sets 54
- single-byte character sets (SBCS), *See* SBCS
- Slavic character set 135
- SMTP character sets 54
- sorting
 - binary 93, 96
 - See Also* binary sorting
 - characters with diacritics 94
 - database-level SQL ORDER BY 95
 - ideographic languages 94
 - PeopleTools in-memory 96
 - performing in COBOL/SQR 98
 - setting sort orders 97
 - special characters 94
 - understanding 93, 94
- specified time zones 26
- spell check, selecting language for 204, 205
- SPF Viewer 122
- SQL
 - database-level SQL ORDER BY sorts 95
 - forcing binary sorts in 97
 - setting column lengths in dynamic 81
- SQR
 - PeopleSoft 107
 - See Also* SQR for PeopleSoft
 - running SWPAUDIT.SQR 159
 - sorting 99
- SQR for PeopleSoft
 - checking encoding settings 122
 - configuring to process international text 112
 - formatting report dates/times 108
 - measuring string length 123
 - printing for A4 paper 107
 - printing international text 113
 - printing limitations for international text 121
 - PSSQR files 112
 - See Also* PSSQR files
 - setting currency precision 108
 - setting up files for PCL/line printing 118
 - setting up files for PostScript printing 119
 - setting up fonts 114
 - setting up PDF fonts 116
 - Strings table enabled SQR (sample) 111
 - supporting character sets 124
 - translating reports 109
 - understanding output types 125
 - using the Append_Report_Translation procedure 110
 - using the Get_Field_Information procedure 110
 - using the Init_Report_Translation procedure 109
 - using the OPEN command 124
 - working with international text 122
- SQR Open command 124
- SQR OPEN command 113, 124
- \$sqr-encoding system variable 122
- \$sqr-encoding-console system variable 122
- \$sqr-encoding-database-api system variable 122
- \$sqr-encoding-file-input system variable 122
- \$sqr-encoding-file-output system variable 122, 124
- \$sqr-encoding-report-output system variable 122
- \$sqr-encoding-source system variable 122
- SQRTRANS.SQC program 109
- status
 - page search 180
 - Replace process 184, 192

- string definitions
 - evaluating PS/nVision variables 105
 - printing/formatting SQR output for multiple languages 109
- string variables
 - evaluating PS/nVision 105
 - using for PS/nVision 137
- strings
 - characters 123
 - See Also* character strings
 - definitions 105
 - See Also* string definitions
 - setting up criteria for PS/nVision 138
 - string functions 59
 - Strings table 104
 - See Also* Strings table
 - translating 242
 - variables 137
 - See Also* string variables
- Strings table
 - enabled SQR (sample) 111
 - evaluating PS/nVision variables 105
 - Strings Table utility 138
 - translating SQRs 109
 - translating strings 242
 - understanding 104
 - using the Append_Report_Translation procedure 110
 - using the Get_Field_Information procedure 110
- style sheets 52
- Substr() function 123
- Substrp() function 123
- Substrt() function 124
- suggestions, submitting xvii
- SWAP_BASE_LANGUAGE
 - command 158, 161
- SWPAUDIT.SQR 159
- Sybase
 - supporting Unicode 40
 - understanding field length semantics 56
- Symbol character set 135
- SYSDATE system variable 25
- system variables
 - %BaseTimeZone 32, 33
 - checking encoding settings 122
 - %clienttimezone 33
 - %ClientTimeZone 32
 - %Language 13

- %Language_Base 13
- \$sqr-encoding-file-output 124
- SYSDATE 25

T

- tables
 - base 150
 - See Also* base tables
 - Currency Code 61
 - International Field Sizes 65
 - PSLANGUAGES 15
 - See Also* PSLANGUAGES table
 - PSPNLFIELD 161
 - PSTZOFFSET 29
 - related language 149
 - See Also* related language tables
 - Strings 104
 - See Also* Strings table
 - Translate 202
 - See Also* Translate table
 - understanding base language 149
- Terminology Management tool
 - choosing languages to search 183
 - defining search text 185
 - defining the search scope 184
 - enabling search processing 183
 - granting search privileges 181
 - objects, modifying searchable 197
 - objects, selecting search 196
 - objects, viewing searchable 196
 - Replace process 179
 - See Also* Replace process
 - running searches 188
 - search criteria, cloning 187
 - search criteria, defining 182
 - search criteria, reviewing 195
 - search criteria, sample 187
 - search results, reviewing 189
 - search results, viewing all 195
 - search results, viewing/replacing 188
 - search status 180
 - search-only results, viewing 195
 - searching 178
 - understanding 177
 - Undo process 179
 - See Also* Undo process
 - upgrade considerations 197
- terms 269
- text

- adding text property files for
 - languages 206
- bi-directional 121
 - See Also* bi-directional text
- defining for searches 185
- international 112
 - See Also* international text
- Text Search Records page 196
- Thai character set 135
- time fields
 - handling in the COBOL Conversion utility 75
 - understanding time zone display settings 31
- Time Zone Data page
 - Daylight Savings tab 29
 - Time Zone tab 28
- Time Zone Offset Generation page 29
- time zones
 - base 25
 - See Also* base time zones
 - maintaining 28
 - setting daylight saving time properties 29
 - setting display options when developing PeopleSoft application objects 31
 - setting in page field properties 33
 - setting in record field properties 32
 - understanding 25
 - understanding display settings 31
 - viewing daylight savings time start/end dates 30
 - viewing definitions 28
- Tiro Typeworks 52
- TLSINST1 utility 65
- TOOLBIN environment variable 213
- %toolbin% environment variable 214
- Translate Application Data page 245
- Translate Messages page 226
- Translate Mobile Page Items page 235
- Translate Mobile Page Names page 234
- Translate Query utility 241
- Translate table
 - adding language codes 202
 - translating values 219, 225
- Translation Designer
 - copying translations to Excel 223
 - entering translations 222
 - expanding/collapsing nodes 221
 - navigating 222
 - opening 218
 - resizing columns/rows 221
 - searching/replacing 222
 - translating review page text 244
 - understanding 217
 - using Application Designer features 223
 - using the information bar 219
 - viewing/hiding rows 221
- translation pages 224
- translations
 - adding to existing databases 156
 - designing global-ready pages 144
 - developing applications 143
 - installing PeopleSoft-provided 155
 - locating resource directories 211
 - Multi Language Entry 170, 171
 - See Also* Multi Language Entry
 - PS/nVision 139
 - required files and tools 209
 - SQR for PeopleSoft 109
 - storing in related language tables 153
 - Terminology Management tool 177
 - See Also* Terminology Management tool
 - Translate table 225
 - See Also* Translate table
 - translating application data 245
 - translating Application Engine definitions 243
 - translating business interlinks 232
 - translating business processes 230
 - translating component interface messages 231
 - translating folder tabs 229
 - translating HTML definitions 247
 - translating images 216
 - translating menu definitions 227
 - translating menu items 228
 - translating message channels 232
 - translating message definitions 243
 - translating messages 226
 - translating mobile data 234
 - translating mobile page items 235
 - translating objects 215
 - translating pages 225
 - translating portals 237
 - translating Process Scheduler definitions 232
 - translating properties files 215

- translating queries 241
- translating record definitions 240
- translating resource files 212
- translating review page text 244
- translating strings 242
- translating trees 238
- Translation Designer 217
 - See Also* Translation Designer
- understanding 209
- understanding custom 199
- understanding options 217
- understanding related language
 - tables 150
- understanding selective 143
- upgrading customizations to multilingual
 - databases 157
- using alternate-language DLLs 210
- using MAKEALTL.BAT 213, 214
- using the Append_Report_Translation
 - procedure 110
- using the Get_Field_Information
 - procedure 110
- using the Init_Report_Translation
 - procedure 109
- trees, translating 238
- Turkish character set 135
- typographical conventions xv

U

- UCS encoding 38, 40, 41, 67
- Undo Messages page 193
- Undo process
 - enabling/disabling 193
 - running 194
 - understanding 179
 - undoing replacements 193
- Unicode
 - architecture/support 43
 - binary sorting, using 96
 - binary sorts in SQL, forcing 97
 - character sets, application server 51
 - character sets, converting 44
 - character sets/fonts in PeopleSoft Pure
 - Internet Architecture, managing 52
 - COBOL Conversion utility 70
 - See Also* COBOL Conversion utility
 - COBOL, storage requirements for 68
 - databases, supporting in 44
 - databases, using Unicode-encoded 50
 - encoding (examples) 41

- encodings 39
- encodings for PeopleSoft COBOL 67
- encodings, supported 136
- error messages 69
- identifying ANSI vs Unicode data 81
- operations that don't support 201
- parsing strings 69, 83
- printing Chinese, Japanese, and Korean
 - PDF text 117
- printing limitations in SQR for
 - PeopleSoft 121
- processing international text 112
- saving properties files 207
- setting column lengths in dynamic
 - SQL 81
- understanding 38
- understanding field length
 - semantics 55
- universal coordinated time (UTC) 25
- Universal Multiple-Octet Coded Character
 - Set (UCS) 38
- UNIX
 - adding translations to databases 156
 - compiling COBOL 71
 - determining character sets 201
 - locating language resources 211
 - PSSQR.UNX file 112
 - See Also* PSSQR files
 - selecting default character sets 51
 - setting up files for PostScript
 - printing 120
 - using MAKEUNIX.BAT 214
- upgrades
 - adding translations 156
 - changing terminology 197
 - upgrading customizations to multilingual
 - databases 157
- US-ASCII, *See* ASCII
- user profile language preferences 14
- User Profiles page 14
- User Search Options page 181
- users
 - granting search privileges 181
 - understanding user time zones 26
- UTC 25
- UTF-16 encoding 40, 41
- UTF-32 encoding 40, 41
- UTF-8 encoding
 - running COBOL 67
 - understanding 40, 41

utilities

- COBOL Conversion 70
 - See Also* COBOL Conversion utility
- International Field Size 65
- native2ascii 207
- Strings Table 138
- Translate Query 241
- translation 229

V

variables

- environment 213
 - See Also* environment variables
- PS/nVision 140
 - See Also* PS/nVision variables
- string 137
 - See Also* string variables
- system 13
 - See Also* system variables
- Verity locales 205
- Vietnamese character set 136
- visual cues xvi

W

warnings xvi

- BIND/SELECT log 79
- COBOL conversion logs 79
- exception date log 80
- exception log 79

web servers

- bypassing the signon page 14
- modifying the signon page 206
- translating properties files 215

Windows

- adding languages to Configuration Manager 205
- adding translations to databases 156
- character sets for CSV queries, selecting 205
- character sets, selecting default 51
- character sets, supporting Unicode 40
- character sets/languages, managing 53
- compiling COBOL 71
- copying translations from Translation Designer 223
- locating language resources 211
- PS/nVision path search feature 139
- PSSQR.INI file 112
 - See Also* PSSQR files

- setting regional options 23
- setting up files for PostScript printing 120
- setting up user-defined fonts 116
- setting/changing language preferences 17
- supporting IMEs 54
- using MAKEALTL.BAT 213

X

- XLAT values, *See* Translate table