

# PeopleSoft®

---

## PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Tables and Business Views

---

**August 2005**

PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Tables and Business Views  
SKU E1\_TOOLS895TTC-B 0805  
Copyright © 2005, Oracle. All rights reserved.

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

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

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

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

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

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

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

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

### **Open Source Disclosure**

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

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright © 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.

# Contents

## General Preface

- About This PeopleBook Preface .....xi**
- PeopleSoft Application Prerequisites.....xi
- PeopleSoft Application Fundamentals.....xi
- Documentation Updates and Printed Documentation.....xii
  - Obtaining Documentation Updates.....xii
  - Ordering Printed Documentation.....xii
- Additional Resources.....xiii
- Typographical Conventions and Visual Cues.....xiv
  - Typographical Conventions.....xiv
  - Visual Cues.....xv
  - Country, Region, and Industry Identifiers.....xvi
  - Currency Codes.....xvi
- Comments and Suggestions.....xvi
- Common Elements Used in PeopleBooks.....xvii

## Preface

- PeopleSoft EnterpriseOne Tools Development Tools: Tables and Business Views Preface.....xix**
- Development Tools: Tables and Business Views Companion Documentation.....xix

## Part 1 Designing Tables in PeopleSoft EnterpriseOne

### Chapter 1

- Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Tables and Business Views.....3**
- Development Tools: Tables and Business Views Overview.....3
- Development Tools: Tables and Business Views Implementation.....3
  - Tables and Business Views Implementation Steps.....3

**Chapter 2**

**Understanding Tables.....5**  
 PeopleSoft EnterpriseOne Table Design Aid.....5

**Chapter 3**

**Designing Tables.....7**  
 Understanding Table Creation.....7  
 Creating Tables.....7  
     Audit Trail Information.....8  
     Table Names.....8  
     Indices Guidelines.....8  
 Working with Table Design Aid.....9  
     Table Design Aid Forms.....9  
     Data Items.....9  
     Indices.....10  
     Table Information.....10  
 Working with Tables.....10  
     Table Generation.....11  
     Index Generation.....11  
     Header File Generation.....11  
     Table Copying.....11  
     Table Removal.....11  
 Viewing the Data in Tables.....12  
 Designing Custom Tables.....12  
     Forms Used to Create Tables.....13  
     Adding Table Objects.....13  
     Modifying Table Objects.....15  
     Selecting Data Items for Tables.....15  
     Defining Indices.....15  
     Previewing Tables.....15  
     Generating Tables.....16  
     Generating Indices.....16  
     Generating Header Files.....17  
     Copying Tables.....17  
     Removing Tables.....17  
     Viewing the Data in Tables.....18

**Chapter 4**

**Working with Table I/O.....19**

Understanding Table I/O.....19

    Available Operations.....19

    Valid Mapping Operators.....20

Understanding Buffered Inserts.....20

    Buffered Insert Error Messaging.....21

Understanding Special Operations.....21

Understanding Handles.....22

    Using Handles.....22

Working With Table I/O.....23

    Prerequisites.....23

    Creating Table I/O Event Rules.....23

    Using Buffered Inserts in Table I/O.....24

    Using Special Operations for Buffered Inserts.....24

    Creating Handles.....24

    Using Handles with Table I/O.....26

**Chapter 5**

**Creating Table Event Rules.....27**

Understanding Table Event Rules.....27

Working with Table Event Rules.....28

    Understanding Table Event Rule Triggers.....28

    Adding Table Event Rules to Table Conversions.....28

**Part 2**  
**Using Cross Reference Facility**

**Chapter 6**

**Understanding Cross Reference Facility.....33**

Understanding the Cross Reference Facility.....33

Searching for PeopleSoft EnterpriseOne Objects.....33

    Searching for Objects.....33

Viewing Field Relationships.....37

    Viewing Field Relationships.....37

Rebuilding Cross Reference Information.....37

    Rebuilding Information for the Cross Reference Facility.....37

Working with the Cross Reference Facility.....	38
Forms Used to Work with the Cross Reference Facility.....	38
Searching for Objects.....	38
Searching for Event Rules.....	39
Viewing Field Relationships.....	39
Rebuilding Cross Reference Information.....	40

## **Part 3 Designing Business Views**

### **Chapter 7**

<b>Understanding Business Views.....</b>	<b>45</b>
Business Views.....	45
Table Joins.....	45
Table Union.....	47
Select Distinct.....	47
Primary Key Fields.....	48

### **Chapter 8**

<b>Designing Business Views.....</b>	<b>49</b>
Understanding Business View Design.....	49
Understanding Business View Naming Conventions.....	49
Understanding Business View Design Aid.....	50
Selecting Tables for Business Views.....	51
Selecting Data Items for Business Views.....	51
Using Select Distinct.....	52
Creating Table Joins.....	54
Creating Table Unions.....	54
Creating Business Views.....	55
Forms Used to Create Business Views.....	55
Adding Business Views.....	55
Selecting Tables for Business Views.....	56
Selecting Data Items for Business Views.....	56
Using Select Distinct.....	56
Creating Table Joins.....	57
Creating Table Unions.....	57

# Part 4 Converting Tables

## Chapter 9

<b>Understanding Table Conversion.....</b>	<b>61</b>
Table Conversions.....	61
Types of Tables to Use.....	62
Business Views in Table Conversions.....	62
Text Files in Table Conversions.....	62
Sort and Selection Criteria in Table Conversions.....	63
Input and Output Environments.....	63

## Chapter 10

<b>Setting Up Table Conversions.....</b>	<b>65</b>
Understanding Table Conversion Setup.....	65
Understanding the Table Conversion Process Flow.....	65
Starting the Table Conversion Director.....	67
Understanding the Navigation Assistant.....	67
Prerequisites.....	67
Forms Used to Add New Table Conversions.....	68
Adding New Table Conversions.....	68
Starting the Table Conversion Director.....	69
Converting Data.....	69
Understanding Data Conversion.....	69
Defining External Data for Table Conversions.....	70
Defining Input and Output Environments for Table Conversions.....	71
Defining Table Conversion Input.....	71
Defining Table Conversion Output.....	73
Mapping Inputs to Outputs.....	74
Specifying Table Conversion Logging Options.....	75
Reviewing Table Conversion Results.....	76
Defining File Formats for Table Conversion Input or Output.....	76
Understanding User Defined Formats.....	77
Prerequisite.....	78
Defining Delimited File Formats with a Single Format.....	78
Defining Delimited File Formats with Multiple Formats.....	79
Defining Fixed-Width File Formats with a Single Format.....	80
Defining Fixed-Width File Formats with Multiple Formats.....	81

Copying Data.....82

    Understanding Copying Data.....82

    Prerequisites.....82

    Defining Data Copy Actions.....83

Copying Data with Table Input.....84

    Understanding Data Copy with Table Input.....84

    Prerequisites.....85

    Defining Copy with Table Input Actions.....85

Deleting Groups of Records.....86

    Understanding Batch Deletion.....86

    Prerequisites.....94

    Defining Batch Deletion Actions.....95

Using Event Rules in Table Conversions.....96

    Understanding Event Rules in Table Conversions.....96

    Creating Event Rules in Table Conversions.....97

**Chapter 11**

**Running Table Conversions.....99**

Understanding Table Conversion Processing.....99

Submitting Table Conversions.....99

    Understanding Submitting Table Conversions.....99

    Understanding Overriding Table Conversion Properties.....100

    Forms Used to Submit Table Conversions.....101

    Overriding Table Conversion Properties at Runtime.....101

    Overriding Table Conversion Properties of New Versions.....102

    Overriding Table Conversion Locations.....102

    Running Table Conversions on the Web Client.....103

Testing Table Conversions.....103

    Understanding Table Conversion Testing.....103

    Setting the Trace Level for Debug Logging on the Workstation.....105

    Setting the Trace Level for Debug Logging on the Server.....105

    Forcing Row-By-Row Conversion.....105

**Chapter 12**

**Preparing Foreign Tables for Table Conversion.....107**

Understanding Foreign Tables.....107

Adding PeopleSoft EnterpriseOne Data Sources.....107

    Forms Used to Add PeopleSoft EnterpriseOne Data Sources.....108

- Adding PeopleSoft EnterpriseOne Data Sources.....108
- Adding PeopleSoft EnterpriseOne Environments.....110
  - Forms Used to Add PeopleSoft EnterpriseOne Environments.....110
  - Adding PeopleSoft EnterpriseOne Environments.....110
- Setting Up Default OCM Mappings.....111
  - Understanding OCM Mapping.....111
  - Forms Used to Set Up Default OCM Mappings.....112
  - Setting Up Default Mappings.....112
  
- Glossary of PeopleSoft Terms.....115**
  
- Index .....125**



# About This PeopleBook Preface

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

This preface discusses:

- PeopleSoft application prerequisites.
- PeopleSoft application fundamentals.
- Documentation updates and printed documentation.
- Additional resources.
- Typographical conventions and visual cues.
- Comments and suggestions.
- Common elements in PeopleBooks.

---

**Note.** PeopleBooks document only page elements, such as fields and check boxes, 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.

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

You should be familiar with navigating the system and adding, updating, and deleting information by using PeopleSoft menus, and pages, forms, or windows. 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 applications.

---

**Note.** Application fundamentals PeopleBooks are not applicable to the PeopleTools product.

---

For some applications, additional, essential information describing the setup and design of your system appears in a companion volume of documentation called the application fundamentals PeopleBook. Most PeopleSoft product lines have a version of the application fundamentals PeopleBook. The preface of each PeopleBook identifies the application fundamentals PeopleBooks that are associated with that PeopleBook.

The application fundamentals PeopleBook consists of important topics that apply to many or all PeopleSoft applications across one or more product lines. 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 the appropriate application fundamentals PeopleBooks. They provide the starting points for fundamental implementation tasks.

---

## Documentation Updates and Printed 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 [peoplebookspres@mmapartner.com](mailto:peoplebookspres@mmapartner.com).

**See Also**

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

---

## Additional Resources

The following resources are located on the PeopleSoft Customer Connection website:

Resource	Navigation
Application maintenance information	Updates + Fixes
Business process diagrams	Support, Documentation, Business Process Maps
Interactive Services Repository	Interactive Services Repository
Hardware and software requirements	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation & Software, Hardware and Software Requirements
Installation guides	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation & Software, Installation Guides and Notes
Integration information	Implement, Optimize + Upgrade, Implementation Guide, Implementation Documentation and Software, Pre-built Integrations for PeopleSoft Enterprise and PeopleSoft EnterpriseOne Applications
Minimum technical requirements (MTRs) (EnterpriseOne only)	Implement, Optimize + Upgrade, Implementation Guide, Supported Platforms
PeopleBook documentation updates	Support, Documentation, Documentation Updates
PeopleSoft support policy	Support, Support Policy
Prerelease notes	Support, Documentation, Documentation Updates, Category, Prerelease Notes
Product release roadmap	Support, Roadmaps + Schedules
Release notes	Support, Documentation, Documentation Updates, Category, Release Notes

Resource	Navigation
Release value proposition	Support, Documentation, Documentation Updates, Category, Release Value Proposition
Statement of direction	Support, Documentation, Documentation Updates, Category, Statement of Direction
Troubleshooting information	Support, Troubleshooting
Upgrade documentation	Support, Documentation, Upgrade Documentation and Scripts

---

## 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
<b>Bold</b>	Indicates PeopleCode function names, business function names, event names, system 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.

Typographical Convention or Visual Cue	Description
“ ” (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 ( ).
[ ] (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.

### 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.

---

## 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](mailto:doc@peoplesoft.com).

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

---

## Common Elements Used in PeopleBooks

<b>Address Book Number</b>	Enter a unique number that identifies the master record for the entity. An address book number can be the identifier for a customer, supplier, company, employee, applicant, participant, tenant, location, and so on. Depending on the application, the field on the form might refer to the address book number as the customer number, supplier number, or company number, employee or applicant id, participant number, and so on.
<b>As If Currency Code</b>	Enter the three-character code to specify the currency that you want to use to view transaction amounts. This code allows you to view the transaction amounts as if they were entered in the specified currency rather than the foreign or domestic currency that was used when the transaction was originally entered.
<b>Batch Number</b>	Displays a number that identifies a group of transactions to be processed by the system. On entry forms, you can assign the batch number or the system can assign it through the Next Numbers program (P0002).
<b>Batch Date</b>	Enter the date in which a batch is created. If you leave this field blank, the system supplies the system date as the batch date.
<b>Batch Status</b>	Displays a code from user-defined code (UDC) table 98/IC that indicates the posting status of a batch. Values are: <i>Blank</i> : Batch is unposted and pending approval. <i>A</i> : The batch is approved for posting, has no errors and is in balance, but it has not yet been posted. <i>D</i> : The batch posted successfully. <i>E</i> : The batch is in error. You must correct the batch before it can post. <i>P</i> : The system is in the process of posting the batch. The batch is unavailable until the posting process is complete. If errors occur during the post, the batch status changes to E. <i>U</i> : The batch is temporarily unavailable because someone is working with it, or the batch appears to be in use because a power failure occurred while the batch was open.
<b>Branch/Plant</b>	Enter a code that identifies a separate entity as a warehouse location, job, project, work center, branch, or plant in which distribution and manufacturing activities occur. In some systems, this is called a business unit.
<b>Business Unit</b>	Enter the alphanumeric code that identifies a separate entity within a business for which you want to track costs. In some systems, this is called a branch/plant.
<b>Category Code</b>	Enter the code that represents a specific category code. Category codes are user-defined codes that you customize to handle the tracking and reporting requirements of your organization.
<b>Company</b>	Enter a code that identifies a specific organization, fund, or other reporting entity. The company code must already exist in the F0010 table and must identify a reporting entity that has a complete balance sheet.

<b>Currency Code</b>	Enter the three-character code that represents the currency of the transaction. PeopleSoft EnterpriseOne provides currency codes that are recognized by the International Organization for Standardization (ISO). The system stores currency codes in the F0013 table.
<b>Document Company</b>	<p>Enter the company number associated with the document. This number, used in conjunction with the document number, document type, and general ledger date, uniquely identifies an original document.</p> <p>If you assign next numbers by company and fiscal year, the system uses the document company to retrieve the correct next number for that company.</p> <p>If two or more original documents have the same document number and document type, you can use the document company to display the document that you want.</p>
<b>Document Number</b>	Displays a number that identifies the original document, which can be a voucher, invoice, journal entry, or time sheet, and so on. On entry forms, you can assign the original document number or the system can assign it through the Next Numbers program.
<b>Document Type</b>	<p>Enter the two-character UDC, from UDC table 00/DT, that identifies the origin and purpose of the transaction, such as a voucher, invoice, journal entry, or time sheet. PeopleSoft EnterpriseOne reserves these prefixes for the document types indicated:</p> <p><i>P</i>: Accounts payable documents.</p> <p><i>R</i>: Accounts receivable documents.</p> <p><i>T</i>: Time and pay documents.</p> <p><i>I</i>: Inventory documents.</p> <p><i>O</i>: Purchase order documents.</p> <p><i>S</i>: Sales order documents.</p>
<b>Effective Date</b>	<p>Enter the date on which an address, item, transaction, or record becomes active. The meaning of this field differs, depending on the program. For example, the effective date can represent any of these dates:</p> <ul style="list-style-type: none"><li>• The date on which a change of address becomes effective.</li><li>• The date on which a lease becomes effective.</li><li>• The date on which a price becomes effective.</li><li>• The date on which the currency exchange rate becomes effective.</li><li>• The date on which a tax rate becomes effective.</li></ul>
<b>Fiscal Period and Fiscal Year</b>	Enter a number that identifies the general ledger period and year. For many programs, you can leave these fields blank to use the current fiscal period and year defined in the Company Names & Number program (P0010).
<b>G/L Date</b> (general ledger date)	Enter the date that identifies the financial period to which a transaction will be posted. The system compares the date that you enter on the transaction to the fiscal date pattern assigned to the company to retrieve the appropriate fiscal period number and year, as well as to perform date validations.

# PeopleSoft EnterpriseOne Tools Development Tools: Tables and Business Views Preface

This preface discusses PeopleSoft EnterpriseOne Development Tools: Table and Business Views companion documentation.

---

## Development Tools: Tables and Business Views Companion Documentation

Additional, essential information describing the setup and design of PeopleSoft EnterpriseOne Tools resides in companion documentation. The companion documentation consists of important topics that apply to PeopleSoft EnterpriseOne Development Tools: Tables and Business Views as well as other PeopleSoft EnterpriseOne Tools. You should be familiar with the contents of these companion PeopleBooks:

- Development Tools: Batch Versions
- Development Tools: Data Dictionary
- Development Tools: APIs and Business Functions
- Development Tools: Data Structure Design
- Development Tools: Event Rules and System Functions
- Configurable Network Computing Implementation

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Batch Versions*, “Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Batch Versions”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Data Dictionary*, “Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Data Dictionary”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: APIs and Business Functions*, “Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: APIs and Business Functions”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Data Structure Design*, “Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Data Structure Design”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Event Rules and System Functions*, “Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Event Rules and Systems Functions”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation*, “Getting Started with PeopleSoft EnterpriseOne Tools Configurable Network Computing Implementation”



## **PART 1**

# **Designing Tables in PeopleSoft EnterpriseOne**

### **Chapter 1**

**Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Tables and Business Views**

### **Chapter 2**

**Understanding Tables**

### **Chapter 3**

**Designing Tables**

### **Chapter 4**

**Working with Table I/O**

### **Chapter 5**

**Creating Table Event Rules**



# CHAPTER 1

## Getting Started with PeopleSoft EnterpriseOne Tools Development Tools: Tables and Business Views

This chapter discusses:

- Development Tools: Tables and Business Views Overview
- Development Tools: Tables and Business Views Implementation

---

### Development Tools: Tables and Business Views Overview

Development Tools: Tables and Business Views addresses three tools from the PeopleSoft EnterpriseOne toolset: Tables Design Aid is used to create and modify tables, Business View Design Aid is used to create and modify business views, and the Table Conversion tool is used to convert tables and copy data between tables.

---

### Development Tools: Tables and Business Views Implementation

This section provides an overview of the steps that are required to implement Development Tools: Tables and Business Views.

#### Tables and Business Views Implementation Steps

This section provides an overview of the steps that are required to implement PeopleSoft EnterpriseOne Tools: Tables and Business Views.

In the planning phase of your implementation, take advantage of all PeopleSoft sources of information, including the installation guides and troubleshooting information. A complete list of these resources appears in the preface in *About These PeopleBooks* with information about where to find the most current version of each.

This table lists the steps for the Development Tools: Tables and Business Views implementation.

<b>Step</b>	<b>Reference</b>
1. Set up permissions to access and use Object Management Workbench (OMW) using Security Workbench.	<i>PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Security Administration</i> , “Using Security Workbench,” Managing Application Security
2. Add yourself to the system in a developer role so that you have permissions to create PeopleSoft EnterpriseOne objects.	<i>PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Object Management Workbench</i> , “Configuring User Roles and Allowed Actions,” Setting Up User Roles
3. Set up permissions to create OMW projects.	<i>PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Object Management Workbench</i> , “Configuring User Roles and Allowed Actions,” Setting Up Allowed User Actions
4. Set up the appropriate database permissions so that you can add tables to the database, drop tables, and add and modify data.	Work with the database administrator to set up these permissions.

## CHAPTER 2

# Understanding Tables

This chapter provides an overview of PeopleSoft EnterpriseOne Table Design Aid.

---

## PeopleSoft EnterpriseOne Table Design Aid

The PeopleSoft EnterpriseOne system uses a relational database.

Database tables store the data that is used by applications in columns and rows. Each column is a data item, and each row is a record. You can create custom tables for use in PeopleSoft EnterpriseOne applications.

You create tables by selecting data items from the data dictionary and assigning key fields as indices. An index enables a database management system (DBMS) to sort and locate records quickly. You must define a table so that the PeopleSoft EnterpriseOne software recognizes that the table exists.

You must use Table Design to generate the table whenever you:

- Create new tables.
- Add or delete data items.
- Add or modify an index.



# CHAPTER 3

## Designing Tables

This chapter provides an overview of table creation and discusses how to:

- Create tables.
- Work with Table Design Aid.
- Work with tables.
- View the data in tables.
- Design custom tables.

---

### Understanding Table Creation

You might be required to create tables for the PeopleSoft EnterpriseOne database to be used by existing or custom applications. You can create tables using Table Design Aid. Access Table Design Aid from Object Management Workbench.

There are several steps in creating tables:

- Create a table object.  
Use the recommended naming conventions when creating tables.
- Design the table.  
Add data dictionary items as table columns and create indices. Include audit fields in the table.
- Generate the table.
- Add data to the table.

---

### Creating Tables

This section discusses:

- Audit trail information.
- Table names.
- Indices guidelines.

## Audit Trail Information

Before you add a new table to the PeopleSoft EnterpriseOne system, determine whether an existing table contains the data items required. If an appropriate table does not exist, you must create a new table.

When you add a new table, you should include these audit trail columns:

- User ID (USER)
- Program ID (PID)
- Machine Key (MKEY)
- Date - Updated (UPMJ)
- Time - Last Updated (UPMT)

## Table Names

Use these naming conventions when adding tables:

The table name can be a maximum of eight characters, and should be formatted as *Fxxxxyyy*, where:

*F* = Data table.

*xx* (second and third digits) = the system code, such as:

- 00 - Foundation environment
- 01 - Address Book
- 03 - Accounts Receivable

*xx* (fourth and fifth digits) = the group type, such as:

- 01 - Master
- 02 - Balance
- 1X - Transaction

*yyy* (sixth through eighth digits) = object version, such as programs that perform similar functions but vary distinctly in specific processing.

LA through LZ = Logical file.

JA through JZ = Table join.

Columns must include a two-character prefix that is used to uniquely identify the table columns. The first character must be alphabetic. The second character can be alphanumeric. You cannot assign special characters (for example, \$, #, or @). The data item alias follows the two-character column prefix. Typically, the column prefix indicates the type of data included in the table; for example, the prefix of the columns in the F0101 (Address Book Master) table begin with *AB*.

A table description can be no more than 60 characters. Ensure that the table description is the topic of the table. If the table description comes from the iSeries, it should be the same name as the file that it represents, such as F0101 (Address Book Master) and F4101 (Item Master).

## Indices Guidelines

If an index includes only one field, use the field name as the index name (for example, Address Number).

If an index includes two fields, list them consecutively; for example, Address Number, Line Number ID.

If an index includes more than two fields, and the first two fields are the same as the first two fields of another index, list the first two fields and follow them by an alpha character; for example, Address Number, Line Number, A. Otherwise, list the fields and follow them by a plus sign (+); for example, Item Number, Branch, +.

Place a comma and space between each index field and between the last index field and the plus sign. Do not include more than 10 fields in an index.

The total length of the index name cannot exceed 19 characters. If you exceed 19 characters, the compiler displays a warning. This situation affects fetches that use the wrong index ID in business functions.

---

## Working with Table Design Aid

This section discusses:

- Table Design Aid forms
- Data items
- Indices
- Table information

### Table Design Aid Forms

Table Design Aid presents these forms within a single window:

- Columns  
Displays the data items that you select to include in the table.
- Data Dictionary Browser  
Use to locate the data items required for the table.
- Indices  
Use to define unique data items for quick sorting and updating of the table.
- Object Properties  
Displays data item attributes for a data item that is selected in the Columns form.

### Data Items

Use data items to create table columns. Table columns store information used by applications. Data items must exist in the data dictionary before you can use them in a table. In Table Design, you can locate data items using the Query By Example (QBE) line of the Data Dictionary Browser. You can double-click data items or drag them to the Columns form to select them. All selected data items are displayed on the Columns form. Tables can contain data items from multiple system codes.

When you modify or delete data items, you must regenerate the table. Changes you make to a table can affect business views and forms that reference that table.

Use the Generate Table option to generate a newly modified table. Generating an existing table clears all data from the table.

---

**Important!** If you delete a table or delete columns from a table then business views that reference that table, or the deleted table columns, are invalid. The system displays error messages when you generate the application.

---

## Indices

Indices are used to locate specific records and to sort records faster. Table indices are like tabs in a card file. Each index is made up of one or more keys, which are individual data items. Use indices to access data in a simple manner, rather than reading the data sequentially.

Tables can have multiple indices; every table must have only one primary index. The primary index is the one unique identifier for each record in the table. Additionally, you can use the primary index to build business views. The system does not allow you to save a table without defining a primary index.

When you modify or delete indices, you must regenerate the table. Changes you make to a table can affect business views and forms that reference that table.

## Table Information

You can use Table Design to review information regarding a table. You can review the table information online or send it to a printer:

- Description  
The description of each data item included in the table.
- Type  
The field type of each data item included in the table.
- Len  
The field length of each data item included in the table.
- Alias  
The alias of each data item included in the table.
- Data Item  
The name of each data item included in the table.
- Table Column Prefix  
The prefix of the table columns.
- Indices  
Information regarding all of the indices included in the table.

---

## Working with Tables

This section discusses:

- Table generation
- Index generation
- Header file generation

- Table copying
- Table removal

## Table Generation

You must generate a table to create a physical table in the database. The table is created based on the specifications that you defined. After the table is generated you can add data to the table. Table generation also creates a .H file, or header file, that is used in business functions and table event rules.

Object Management Workbench provides a central location from which you can manage tables.

After you have selected data items for a table and assigned indices, you can configure the table for a specific data source. Object Management Workbench employs the Object Configuration Manager application (P986110) to configure tables. You can configure the table within any existing data source. If you do not indicate a data source, the software automatically configures the table according to the default data source mapping. You can change the path code to generate tables in a different location. Doing so causes the system to perform a drop statement, similar to the remove table, after which the system recreates the table.

You must regenerate a table after you modify it. If you regenerate an existing table, the table data is lost. To ensure that data is not lost, you must export the data, generate the table, and then copy the data back into the generated table.

## Index Generation

When you create additional indices or modify existing indices, you must regenerate the indices. Doing so modifies the .H file, but you do not lose existing data as you do when you regenerate the entire table.

## Header File Generation

Header files, or .H files, are used in business functions and table event rules. Occasionally, you might run across a table that does not include a header file. You can generate header files without having to generate the entire table.

Header files are located in the Include folder in the path code in which the file was generated. For example, E811\DV811\Include.

## Table Copying

Using the Copy Table option, you can copy tables from one data source to another. Doing so does not copy the table specifications. You can also use Table Conversion to copy tables from one data source to another.

## Table Removal

You cannot physically delete a table using Table Design. If you delete a table from Table Design, the system deletes only the specifications; it does not delete the physical table.

To completely remove a table from the system, you must use the Remove Table From Database option.

---

## Viewing the Data in Tables

You can view data in tables using Data Browser on the web client or Universal Table Browser on the Microsoft Windows client. Both tools enable you to verify the existence of data in tables, as well as determine the structure of the table. Both tools can be used to view data in all PeopleSoft EnterpriseOne supported databases, independent of the type of database that you access.

Universal Table Browser is an executable application that is part of the PeopleSoft EnterpriseOne Microsoft Windows client install. You cannot use PeopleSoft EnterpriseOne security to control user permissions for the Universal Table Browser. However, you can apply form security to the Table and Data Source Selection form (W98TAMC). This action secures the Universal Table Browser because the executable cannot function without this form. All column and row security that you set up through Security Workbench applies to the Universal Table Browser.

Data Browser is part of the PeopleSoft EnterpriseOne web client product. You can set up security for the Data Browser using Security Workbench.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: System Administration*, “Using the Universal Table Browser”

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Foundation*, “Using the PeopleSoft EnterpriseOne Web Application User Interface,” Viewing the Data in Tables and Business Views

---

## Designing Custom Tables

This section discusses how to:

- Add table objects.
- Modify table objects.
- Select data items for tables.
- Define indices.
- Preview tables.
- Generate tables.
- Generate indices.
- Generate header files.
- Copy tables.
- Remove tables.
- View the data in tables.

## Forms Used to Create Tables

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	EnterpriseOne Life Cycle Tools, Application Development (GH902), Object Management, Object Management Workbench	Select and view objects in projects and access the PeopleSoft EnterpriseOne design tools.
Add EnterpriseOne Object to the Project	W98220C	Click the Objects node of a project and click Add on the Object Management Workbench form.	Add a new object to a project.
Add Object	W9861AF	Select Table and click OK on the Add EnterpriseOne Object to the Project form.	Create a table.
Table Design Aid	W9860AL	Complete the object information and click OK on the Add Object form.	Access Table Design Aid, generate header files, generate tables, generate indexes, remove tables from the database, and copy tables.
Generate Table	W9866E	Select the Table Operations tab, and click Generate Table on the Table Design form.	Generate tables.
Generate Indexes	W9866J	Select the Table Operations tab, and click Generate Indexes on the Table Design form.	Generate indices.
Copy Table	W9866M	Select the Table Operations tab, and click Copy Table on the Table Design form.	Copy tables.
Remove Table	W9866D	Select the Table Operations tab, and click Remove Table from Database on the Table Design form.	Remove tables.
Universal Table Browser	N/A	EnterpriseOne Life Cycle Tools, Application Development Tools (GH902), Object Management	View the data in tables.

## Adding Table Objects

Access the Add Object form.

The screenshot shows a window titled "Object Management Workbench - [Add Object]". The window has a menu bar with "File", "Edit", "Preferences", "Window", and "Help". Below the menu bar is a toolbar with icons for "OK", "Cancel", "Dismiss", "Apply", "Links", "Display", "OLE", and "Internet". The main area of the window is a form with the following fields:

- Object Name:** F550101A
- Object Type:** TBLE
- Description:** Address Book
- Product Code:** 55 (Reserved for Clients)
- Product System Code:** 01 (Address Book)
- Column Prefix:** AB
- Object Use:** 2 (Files)

Add Object form

- Object Name** A unique name for the table. Use the recommended naming convention for naming PeopleSoft EnterpriseOne objects.
- Description** A meaningful description of the table.
- Product Code** A user defined code (98/SY) value. Use one of the values from the range of values reserved for clients, 55–59.
- Product System Code** A user defined code (98/SY) value that represents the PeopleSoft EnterpriseOne system where the data is used. This value is used for reporting and jargon purposes. Example values include:
- 01:* Address Book
  - 03B:* Accounts Receivable
  - 04:* Accounts Payable
  - 09:* General Accounting
  - 11:* Multicurrency
- Object Use** A value that indicates the use of the object. For example, the object may be used to create a program, a master file, or a transaction journal. A 2, or any other value from the 200 series, represents tables.
- Object Type** An abbreviation that identifies the type of object being created. The object type for a table is TBLE. This is a display field that is populated by the system based on the type of object that you selected on the Add Object form.
- Column Prefix** The two-character prefix used for the table column names.

## Modifying Table Objects

Access the Table Design form.

1. Select the Summary tab and revise these fields as appropriate:
  - Description
  - Product Code
  - Product System Code
  - Object Use
2. To document the table, select the Attachments tab, and then enter the information in the text area.

## Selecting Data Items for Tables

Access the Table Design form.

1. Select the Design Tools tab, and click Start Table Design Aid.
2. On the Data Dictionary Browser form, use the QBE line to locate the required data dictionary items to include in the table.
3. Drag each required data dictionary item from the Data Dictionary Browser to the Columns form.
4. To remove a column from a table, select the column and select Delete from the Edit menu.

## Defining Indices

Access a table in the Table Design Aid form.

1. Click the indices form so that it is active and the indices menu is visible.
2. From the indices menu, select Add New.

You can also drag indices from the Columns form into the Indices form.

The index description is Untitled and the index is marked with a key icon that displays the letter *P* to indicate a primary index.

3. Enter a name for the index, and press ENTER.
4. On the Columns form, drag appropriate columns to the index.

A unique index is marked with a single key. You can right-click the index and select Unique from the Index menu to toggle the unique status. The Unique Primary Index cannot be changed to a non-unique status.
5. Right-click the data item and select or clear the Ascending option to indicate the sort order of ascending or descending for an index column.

An upward-pointing arrow indicates that the index column is sorted in ascending order.

## Previewing Tables

Access a table in the Table Design Aid form.

1. Click the Columns form so that it is active and select Print Preview from the File menu.

A preview of the table appears on the Columns form in place of the column names.
2. On the Columns form, click Zoom In to enlarge the preview.

3. Click Print to send the information to the printer.

## Generating Tables

Access the Generate Table form.

Generate Table form

<b>Table Name</b>	The unique name of the table. This field is display only and is populated by the system based on the name you gave the table when you created it.
<b>Data Source</b>	The name that identifies the data source.
<b>Object Owner ID</b>	The owner ID of the database in the defined data source.
<b>Password</b>	The database password that corresponds to the owner ID.

## Generating Indices

Access the Generate Indexes form.

<b>Table Name</b>	The unique name of the table. This field is display only and is populated by the system based on the name you gave the table when you created it.
<b>Data Source</b>	The name that identifies the data source.
<b>Object Owner ID</b>	The owner ID of the database in the defined data source.
<b>Password</b>	The database password that corresponds to the owner ID.

## Generating Header Files

Access the Table Design Aid form.

1. Select the Design Tools tab, and click Generate Header File.
2. The system generates a .H, or header, file.

## Copying Tables

Access the Copy Table form.

Copy Table form

<b>Table Name</b>	The unique name of the table. This field is display only and is populated by the system based on the name you gave the table when you created it.
<b>Source Data Source</b>	The name that identifies the source data source to copy from.
<b>Destination Data Source</b>	The name that identifies the target data source to copy to.
<b>Object Owner ID</b>	The owner ID of the database in the defined data source.
<b>Password</b>	The database password that corresponds to the owner ID.

## Removing Tables

Access the Remove Table form.

<b>Table Name</b>	The unique name of the table. This field is display only and is populated by the system based on the name you gave the table when you created it.
-------------------	---

<b>Data Source</b>	The name that identifies the data source.
<b>Object Owner ID</b>	The owner ID of the database in the defined data source.
<b>Password</b>	The database password that corresponds to the owner ID.

## Viewing the Data in Tables

Access the Universal Table Browser form.

<b>Table</b>	The name that identifies a table in PeopleSoft EnterpriseOne. For example, F0101 is the name of the Address Book master table. You can use the visual assist to search for a table
<b>Data Source</b>	The data source in which the table resides. The default value is obtained from the Object Configuration Manager (OCM) settings for the environment in which the user is signed on. Use the visual assist to select from a list of valid PeopleSoft EnterpriseOne data sources.
<b>Format Data</b>	<p>Indicates whether the Universal Table Browser formats data once it is fetched from the database. Data Browser on the web does not provide this option. It always formats data after it is fetched from the database.</p> <p><i>Formatted:</i> The Universal Table Browser formats data according to the specifications of the PeopleSoft EnterpriseOne data dictionary item. For example, assume that the data item PROC is a numeric field with a size of 15, and includes four display decimals. For a PROC value of 56.2185, the Universal Table Browser displays a formatted value (using the data dictionary editing) as 56.2185, even though the value is stored in the database as 562185.</p> <p><i>Unformatted:</i> The Universal Table Browser displays the data according to the specification of the database and the data item type (such as numeric). For example, assume that the table data item PROC is a numeric field stored in the database. Depending on the type of database, the default for this field might be 32 with a precision of 15. Because PeopleSoft EnterpriseOne does not store the decimals in the database, a PROC value of 56.2185 would be stored by the database as 562185.0000000000000000 and displayed the same using Universal Table Browser.</p>

# CHAPTER 4

## Working with Table I/O

This chapter provides overviews of table I/O, buffered inserts, special operations, and handles and discusses how to work with table I/O.

---

### Understanding Table I/O

Use the Table I/O feature in Event Rules Design to create instructions that perform table input and output (I/O). Table I/O enables you to access a table through event rules. You can use table I/O to:

- Retrieve records.
- Update or delete records.
- Add records.

Table I/O event rules enable event rule support for database access.

### Available Operations

Using table I/O, you can perform these operations:

Operation	Description
FetchSingle	Combines Select and Fetch in a basic operation. Indexed columns are used for the Select and non-indexed columns are used for the Fetch. The operation opens a table for input or output but does not close it. All tables that do not use handles close automatically when the form that uses them is closed.
Insert	Inserts a new row.
Update	Updates an existing row. Only those columns mapped (presently in tables with or without handles) are updated. You can do partial key updates with tables and handles to tables. If you do not specify all the keys, then several records might be updated.
Delete	Deletes one or more rows in a table.
Open	Opens a table or business view.
Close	Closes a table or business view.

Operation	Description
Select	Selects one or more rows for a subsequent FetchNext operation.
SelectAll	Selects all rows for a subsequent FetchNext operation.
FetchNext	Fetches rows that you specify. You can fetch multiple records with multiple FetchNext operations or with a FetchNext operation in a loop.

## Valid Mapping Operators

The Mapping form displays available objects that you can map to selected table columns. For SELECT statements, the available objects are used to build a WHERE clause. For FETCH statements, the available objects are used to receive data fetched from the database.

Key columns have an asterisk (\*) next to them.

You can use these operators for mapping specific table I/O operations:

Table I/O Operation	Mapping Operators
FetchSingle	Index Fields: =, <, <=, >, >=, !=, Like Non-Index Fields: Copy Target
Insert	All Fields: Copy Source
Update	Index Fields: = Non-Index Fields: Copy Source
Delete	All Fields: =
Open	N/A
Close	N/A
Select	All Fields: =, <, <=, >, >=, !=, Like
SelectAll	N/A

---

## Understanding Buffered Inserts

You can use buffered inserts to improve performance when you insert hundreds or thousands of records into a single database table and you don't need immediate feedback if an insertion failure occurs. You can use buffered inserts with table conversion, table I/O, batch processes, and business functions but they are not available for interactive applications. Buffered inserts are available only with Oracle (V8 and later), DB2/400, and SQL Server. Buffered inserts are not available with Access, post-insert triggers, or multiple-table views. The PeopleSoft database middleware delivers records to the database management system one buffer load at a time.

When you request buffering, the database records are inserted individually and the buffer is automatically flushed when it fills; that is, the PeopleSoft EnterpriseOne database middleware delivers the buffer to the database management system. The buffer can also be explicitly flushed. For example, the buffer flushes automatically when you commit a transaction or when you close a table or view. The business function, table conversion engine, or table I/O can explicitly request that the buffer be flushed as well.

## Buffered Insert Error Messaging

Because the system provides no immediate feedback if an insertion fails, you should use caution when you decide to use buffered inserts. If an insertion fails, the error appears in the log file. Consequently, buffered inserts are used primarily with batch applications.

Unless you are using the Table Conversion application, you must request more detailed information from the middleware to get detailed error messages. In the Table Conversion application, the table conversion engine automatically performs this task. You can enable tracing to receive more detailed error messages. Otherwise, you get an error message that the insert failed. You should clear the output tables so that you do not receive duplicate error logging.

After you set up buffered inserts, you can use Special Operations to flush the buffers or get error messages.

---

## Understanding Special Operations

In addition to the basic and advanced operations available on the Insert TableIO Operation form, there are two special operations:

- Flush Insert Buffer

To maintain data integrity, you should flush the insert buffer before you perform any operations other than an insert. If you fail to do so, the results of recent inserts might not be reflected in other operations, and the operations might not work properly.

When you use the flush insert buffer option for a specific table, you must flush the buffer before you close the table.

- Get Error Data

To retrieve errors for records that the system did not insert properly, use the Get Error Data option. Depending on when the buffers are flushed, or when you begin another insert, you might overwrite the error information for a specific insert. If error information is critical, retrieve the information before the next insert begins.

If you need to perform special error handling, set it up after each table I/O insert and each Flush Insert Buffer option. Always retrieve the error information *before* you begin the next table operation.

---

**Note.** The return code of each table I/O statement is stored in system variable SV File\_IO\_Status. If an Insert or Flush Insert Buffer operation failed, the SV File\_IO\_Status system variable might contain a value of CO\_ERROR\_DETAILS\_AVAILABLE. If so, you need to run the Get Error Data table I/O special operation to retrieve the error data. The Get Error Data operation returns the values used in the insert for all the requested columns.

---

### Get Error Data Event Rules Example

This example illustrates how you can use CO\_ERROR\_DETAILS\_AVAILABLE.

```
F0101.Insert //Attempt an insert
```

```

SL AgingDaysAP1 -> TK Address Number
RC Page - -> TK Tax ID
SL TargetEnvironment -> TK Description - Compressed
If SV File_IO_Status is equal to CO ERROR_DETAILS_AVAILABLE
F0101.Get Error Data //Failed with errors so get errors
    //Map values used in insert to the
    //specified fields.
SL AgingDaysAP1 <- TK Address Number
RC TESTT <- TK Tax ID
SL ReportName <- TK Description - Compressed
End If

```

---

## Understanding Handles

In PeopleSoft EnterpriseOne, the term *handle* in table I/O operations refers to a type of file pointer. This file pointer connects the application or Universal Batch Engine (UBE) with the PeopleSoft EnterpriseOne database middleware that communicates with the database manager. Handles point to a database table, and they are references to an address within the database middleware. Handles enable you to perform these operations, which cannot be performed using non-handle table I/O operations:

- Concurrently open multiple instances of a single table or business view
- Open a table or business view in an environment other than the environment in which you signed into  
This feature is particularly helpful when you receive an upgrade to PeopleSoft EnterpriseOne software or when you need to convert data from another system into PeopleSoft EnterpriseOne software.
- Pass handles into a form, named event rule, or business function so that you do not need to open a table or business view more than once.

---

**Note.** You cannot use handles in transaction processing.

---

If you pass a handle to a form or a named event rule, the data structure for the form or named event rule must contain a HANDLE data item as a member. In the form interconnect or business function call, you must assign a handle value from your event rules to the HANDLE data structure member. You can use this handle in the form or named event rule in the same way that you use any other table I/O handle.

You must explicitly open and close handles, unlike other table I/O operations in which the system implicitly opens or closes the table for you. You must open a handle before you can use it. All of the table I/O operations except Open work the same for handles as they do for tables or business views. When you are finished using a handle, you must explicitly close it. Close the handle in the same way that you close a table or business view except that you select Handle.

## Using Handles

In order to use handles in table I/O event rules, you must:

- Define the handle in the data dictionary.
- Create a handle variable in event rules.
- Open the handle explicitly.

You might be able to find an existing handle for your table I/O. Whether you use an existing handle or create a new handle, you must create a handle variable in Event Rules Design. Create a handle variable in the same way that you create other variables. You can use any scope that is necessary.

After you create a handle variable, you must explicitly open the handle. Then, after performing the required table I/O, you must explicitly close it.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Data Dictionary*, “Defining a Data Dictionary Item,” Creating a Data Dictionary Item

---

## Working With Table I/O

This section lists the prerequisites and discusses how to:

- Create table I/O event rules.
- Use buffered inserts in table I/O.
- Use special operations for buffered inserts.
- Create handles.
- Use handles with table I/O.

### Prerequisites

Before creating table I/O event rules, ensure that you:

- Create a batch application.
- If you want to update the database, ensure that the batch application is defined as an Update Report.

### Creating Table I/O Event Rules

Open a batch application in Report Design Aid and access Event Rules Design.

1. Click the Table I/O button.
2. On the Insert TableIO Operation form, select an operation from the basic, advanced, or special operations and click Next.
3. On the Data Source form, select a table, business view, or handle as the data source and click Next.
4. On the Mapping form, select fields from the Available Objects list to map to columns in the mapping.

The key field must be mapped. Map as many other fields as required to ensure that the correct data is associated with the correct record.

5. Click the Operation for each map until you locate the operator that you want.  
The default operator is equal.
6. Click Finish to save the operation and return to Event Rules Design.

## Using Buffered Inserts in Table I/O

Open a batch application in Report Design Aid and access Event Rules Design.

1. Click the Table I/O button.
2. On the Insert TableIO Operation form, select Open under Advanced Operations and click Next.
3. On the Data Source form, select the table that you want to use, and click Advanced Options.
4. On the Advanced Options form, select Buffer Inserts and click OK.

## Using Special Operations for Buffered Inserts

Open a batch application in Report Design Aid and access Event Rules Design.

1. At the point at which you want to perform either a Flush Insert Buffer or Get Error Data operation, click the Table I/O button.
2. On the Insert TableIO Operation form, select one of these options under Special Operations.
  - Flush Insert Buffer
  - Get Error Data

## Creating Handles

EnterpriseOne Life Cycle Tools, Application Development, Data Dictionary Design, select Work With Data Dictionary Items to access the Work With Data Items form.

**Work With Data Dictionary Items - [Data Item Specifications]**

File Edit Preferences Form Window Help

OK Can... Dis... Algo Links Descr... OLE ... Internet

**Item Specifications** Item Glossary Default Value Visual Assist Edit Rule Display Rule Next

Data Item: HandleF0116 Alias: HF550116

Glossary Group: D Primary Data Elements:  Item Parent:

**General Information**

Description: Handle for F0116 Product Code: 55

Product Code Reporting: 00

Data Type: 7 Identifier (ID): Size: 11 File Decimals:

Class: HANDLE Display Decimals: 0

Control Type: 4 Item Occurrences:

Row Description: Table Handle F0116

Column Title: Table

Handle F0116

Upper Case Only  Row Security  Allow Blank Entry  Auto Include  Do Not Total

Data Item Specifications form

<b>Data Item</b>	The name of the data item. This is a 32-character alphabetical field that does not allow spaces or special characters.
<b>Alias</b>	A unique identifier.
<b>Glossary Group</b>	A user defined code (H95/DG) that indicates the type of data item. A handle is glossary group D, Primary Data Elements.
<b>Description</b>	A meaningful description of the data item.
<b>Product Code</b>	A user defined code value (98/SY). Use one of the values from the range of values reserved for clients, 55–59.
<b>Product System Code</b>	A user defined code (98/SY) value that represents the PeopleSoft EnterpriseOne system where the data is used. This value is used for reporting and jargon purposes. Example values include: <i>01</i> : Address Book <i>03B</i> : Accounts Receivable <i>04</i> : Accounts Payable <i>09</i> : General Accounting

	<i>II</i> : Multicurrency
<b>Data Type</b>	The style or classification of data. Examples include, string, date, character, and identifier. A handle is an Identifier data type.
<b>Size</b>	The field size of the data item.
<b>Class</b>	Defines the essential attributes and characteristics of the data item. Examples include, name, email, and table handle.
<b>Control Type</b>	Indicates the type of graphical user control that is associated with the data item. Examples include, push button, radio button, user defined code.
<b>Row Description</b>	The description of the data item as it will appear in a group report.
<b>Column Title</b>	The description of the data item as it will appear in a columnar report.

## Using Handles with Table I/O

Open a batch application in Report Design Aid and access Event Rules Design.

1. Create an event rule variable using the appropriate handle data dictionary item.
2. Click the Table I/O button.
3. Select Open under Advanced Operations and click Next.
4. On the Data Source form, select the Handles tab.
5. Select the handle that you want to open and click Next.
6. Select a variable that contains the name of the environment in which you want to open the table and click Finish.

If you want to open the table in the login environment, select the system value *SL LoginEnvironment*. System values also exist for the source and target environments if you use Table I/O in Table Conversion.

7. Click the Table I/O button.
8. Select Close and click Next.
9. On the Data Source form, select the Handles tab.
10. Select the handle that you want to close, and click Finish.

# CHAPTER 5

## Creating Table Event Rules

This chapter provides an overview of table event rules and discusses how to work with table event rules.

---

### Understanding Table Event Rules

Use table event rules to attach database triggers (or programs) that automatically run whenever an action occurs against the table. An action against a table is called an event. When you create a PeopleSoft EnterpriseOne database trigger, you must first determine which event activates the trigger and then use Event Rules Design to create the database trigger.

Table event rules provide embedded logic at the table level. Table event rules have their own location, events, and system functions. When you use table event rules, neither the calling application nor the user is notified of changes or events to the table. No form or report interconnect is available with table event rules.

You can use table event rules for data integrity. For example, when you delete a record in Address Book, you might want to delete all associated records, such as phone and category codes. You can also use table event rules for currency. The *Currency Conversion is On* event rule handles currency information in table event rules.

You can attach event rules on a table-by-table basis to these events:

- After Record is Deleted
- After Record is Fetched
- After Record is Inserted
- After Record is Updated
- Before Record is Deleted
- Before Record is Fetched
- Before Record is Inserted
- Before Record is Updated
- Currency Conversion is On

This table describes the functions that you can use in table event rules:

Button	Description
Assignment\Expression	Creates an assignment or a complex expression.
If\While	Creates an IF/WHILE conditional statement.
Business Function	Attaches an existing business function.

Button	Description
System Function	Attaches an existing PeopleSoft EnterpriseOne system function.
Variables	Creates a Event Rule variable.
Else	Inserts an ELSE clause, which is only valid within the bounds of IF and ENDIF.
Table I/O	Enables event rule support for database access. Performs table input and output, data validations, and record retrieval.

You do not need to create and associate data structures to table event rule functions. The table itself is the data structure that is passed to the table event rule function.

---

## Working with Table Event Rules

This section provides an overview of table event rule triggers and discusses how to add table event rules to table conversions.

### Understanding Table Event Rule Triggers

To create table event rules, you must:

- Create the table trigger in Event Rules Design.
- Generate the PeopleSoft EnterpriseOne table trigger as C code.
- Build the table trigger.

When you build table triggers, the system performs these actions:

- Converts the event rule to C source code, which creates the files OBNM.c and OBNM.hxx (OBNM = Object Name). The source file contains one function per table event rule event.
- Creates a make file to compile the generated code.
- Runs the make file to compile the new functions and to add them into JDBTRIG.DLL. This consolidated DLL contains table event rule functions.

### Adding Table Event Rules to Table Conversions

Access the Object Management Workbench form.

1. Check out the table to which you want to attach event rules, and click Design.
2. On the Table Design form, select the Design Tools tab and click Start Table Trigger Design Aid
3. From the Events list, select an event.
4. On Event Rules Design, click one of the event rule buttons and complete the event rules.
5. Click Save to save the event rule specifications, and then click Close.

6. If you are creating a *new* table in Table Design Aid, select the Table Operations tab and click Generate Table.

---

**Important!** Never perform this step on an existing table because it clears all data.

---

7. On the Generate Table form, complete these fields and click OK:
  - Data Source
  - Password
8. On the Table Design form, select the Design Tools tab, and click Build Table Triggers.
9. To review a log of the build, select Generate Header File, and then open the file that is created by the system.

The creation of the table event rules is complete. The newly created or modified table event rule is called from the PeopleSoft EnterpriseOne database middleware whenever the corresponding event occurs against the table.



## **PART 2**

# Using Cross Reference Facility

### **Chapter 6**

#### **Understanding Cross Reference Facility**



## CHAPTER 6

# Understanding Cross Reference Facility

This chapter provides an overview of the cross reference facility and discusses how to:

- Search for PeopleSoft EnterpriseOne objects.
- View field relationships.
- Rebuild cross reference information.
- Work with the Cross Reference Facility.

---

## Understanding the Cross Reference Facility

You can use the Cross Reference Facility to determine where specific objects are used. You can also view relationships between objects and their components. For example, you can use Cross Reference Facility to:

- Identify each instance in which a business function is used.
- View a list of forms within an application.
- Display all fields within a business view.
- Cross-reference all applications in which a specific field is used.

---

**Warning!** Cross Reference Facility is not automatically updated when you add or modify objects. You must rebuild relationships to include the new information.

---

---

## Searching for PeopleSoft EnterpriseOne Objects

This section discusses searching for objects.

### Searching for Objects

You can search for these objects by object name:

- Data items
- Interactive applications
- Batch applications
- Business functions
- Business views

- Data structures
- Tables
- Forms

### **Searching for Data Items**

You can determine where specific data items are used by searching for:

- Forms that use a data item.
- Universal Batch Engine applications (UBEs) that use a data item.
- UBE event rules that use a data item as a variable.
- Applications that use a data item as a variable.
- Named event rules that use a data item as a variable.
- Functions called by smart field data items.
- Edit rule functions called by a data item.
- Display rule functions called by a data item.
- Search forms used by data items.
- Processing options that use a data item.
- Generic text data structures that use a data item.
- Business function data structures that use a data item.
- All data structures that use a data item.
- Tables that use a data item.
- Indices that use a data item.
- Business views that use a data item.
- Table event rules that use a data item.

### **Searching for Interactive Applications**

You can locate a variety of information about interactive applications:

- Applications that call an application.
- Applications that call a business function.
- Data structures that are used by an application.
- Tables that are used by an application.
- Forms for an application.
- Data items for an application.
- Data items that are used as variables in an application.
- Processing options for an application.
- Business views for an application.

### **Searching for Batch Applications**

You can locate information about UBEs and how they are used:

- UBEs that call a business function.
- Tables that are used by a UBE.
- Data structures that are used by a UBE.
- UBEs that are called by a UBE.
- UBEs that call a UBE.
- Processing options for a UBE.
- Business views for a UBE.
- Data items that are used by a UBE.
- Data items that are used as variables in a UBE.

### **Searching for Business Functions**

You can locate a variety of information about business functions and how they are used:

- The places in which a business function is used.
- Data structures that are used by a named event rule.
- Business functions that are called by an application.
- Business functions that are called by a UBE.
- Tables that are used by a business function.
- Business functions that are called by a business function.
- Business functions that call a business function.
- Data items that are used as variables in a named event rule.
- Tables that are used by a named event rule.
- Business functions that are called by a named event rule.
- Smart field data items that call a column header function.
- The places in which a function is used.
- Smart field data items that call a value business function.
- Data items that call edit rule functions.
- Data items that call display rule functions.
- Source and header files by string.
- Table event rules that use a business function.

### **Searching for Business Views**

You can locate information about business views and how they are used:

- Applications that use a business view.
- UBEs that use a business view.
- Forms that use a business view.
- Data items for a business view.
- Tables for a business view.

## Searching for Data Structures

You can locate a variety of information about data structures and how they are used:

- Applications that use a data structure.
- Business functions that use a data structure.
- UBEs that use a data structure.
- Data items for a processing option.
- Data items for generic text data structures.
- Data items for a business function data structure.
- Data items for all data structures.
- Table event rules that use a data structure.

## Searching for Tables

You can locate a variety of information about tables and how they are used:

- Business functions that use a table.
- Applications that use a table.
- UBEs that use a table.
- Named event rules that use a table.
- Forms that use a table.
- Data items for a table.
- Indices for a table.
- Business views that use a table.
- Data items that are used by table event rules.
- Business functions that are used by table event rules.
- Data structures that are used by table event rules.
- Tables that are used by table event rules.

## Searching for Forms

You can locate a variety of information about forms and how they are used:

- Forms that are called by an application.
- Tables for a form.
- Data items for a form.
- Business views for a form.
- Data items that use a search form.

## Searching for Event Rules

You can search where a particular control is used in the Event Rules of an application.

---

## Viewing Field Relationships

This section discusses viewing field relationships.

### Viewing Field Relationships

The Field Relationships form is meaningful only for these cross-reference search types:

Search Type	Description
DA	Data items that are used by an application
FA	Forms for an application
FI	Forms that use a data item
SA	Data structure for an application

In these instances, the Field Relationships form displays the control type for a field, such as:

- *BC* for a business view column
- *FI* for a form interconnect
- *GC* for a grid control
- *FC* for a form control

---

## Rebuilding Cross Reference Information

This section discusses rebuilding information for the Cross Reference Facility

### Rebuilding Information for the Cross Reference Facility

When developers create and modify objects, cross-reference information can become out-of-date. Because the cross-reference files are not automatically rebuilt when objects are created and modified, you must manually do so periodically. You can also regularly schedule cross-reference builds to ensure that the cross-reference information is get up-to-date.

The Cross Reference application contains multiple forms. Each form contain a grid of cross reference records. Each row contains a date column (the far right column) that indicates on which date the cross reference record was built. Use this date to verify if cross reference records are out of date. If information is out-of-date, use the Rebuild option from any of the cross-reference forms.

Cross-reference builds use relational database tables, not local specifications.

## Working with the Cross Reference Facility

This section discusses how to:

- Search for objects.
- Search for event rules.
- View field relationships.
- Rebuild cross reference information.

### Forms Used to Work with the Cross Reference Facility

Form Name	FormID	Navigation	Usage
Cross Reference	W980011K	EnterpriseOne Life Cycle Tools, Application Development (GH902), Object Management, Cross Reference Facility	Search for objects and rebuild cross reference information.
ER Search	W980011A	Select Event Rules from the Form menu on the Cross Reference form.	Search for event rules.
Field Relationships	W980011B	Select Field Relationships from the Form menu on the Cross Reference form.	View field relationships.

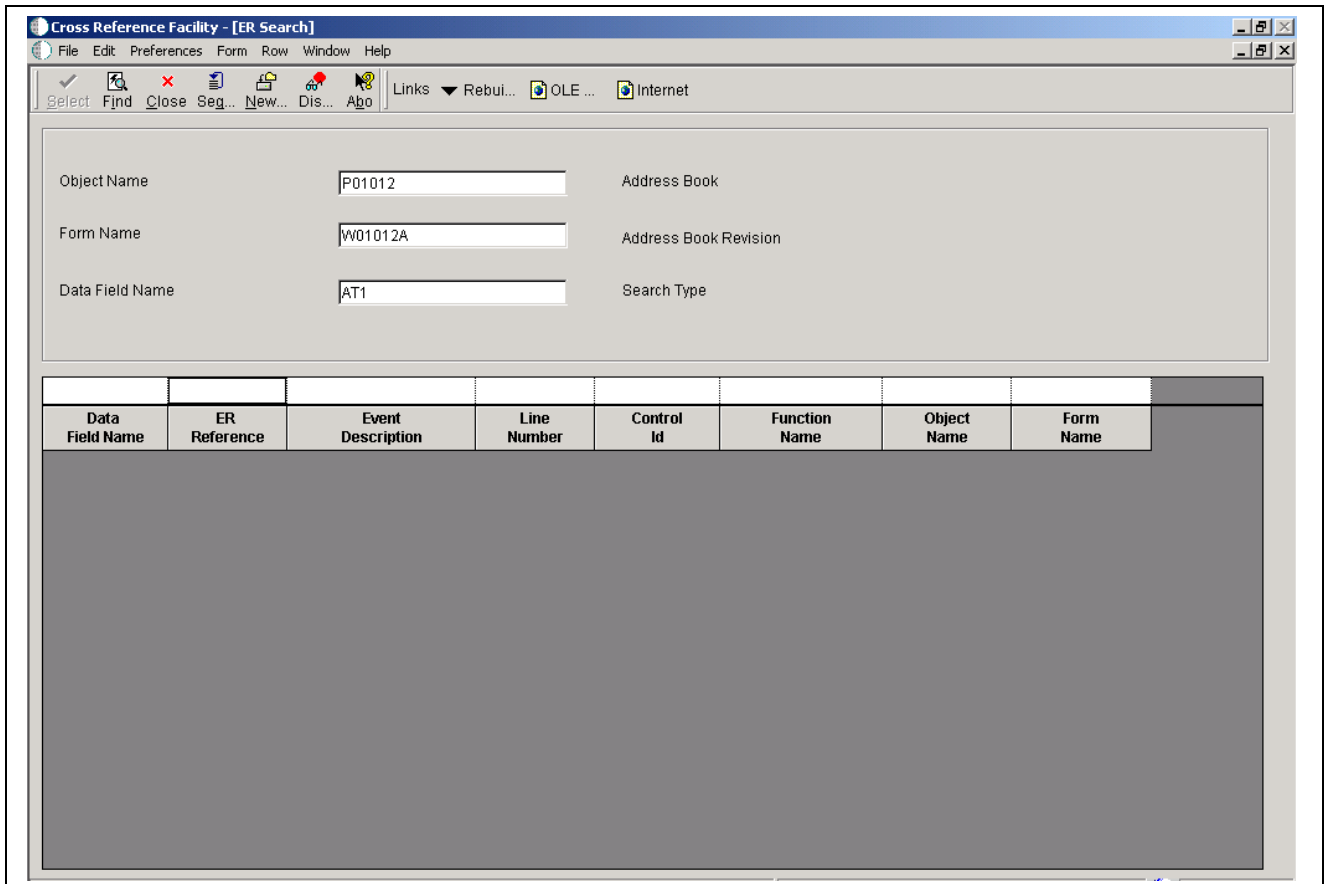
### Searching for Objects

Access the Cross Reference form.

<b>Data Items</b>	Select a search option and from the appropriate form, enter the name of the data item to search.
<b>Interactive Applications</b>	Select a search option and from the appropriate form, enter the name of the interactive application to search.
<b>Batch Applications</b>	Select a search option and from the appropriate form, enter the name of the batch application to search.
<b>Business Functions</b>	Select a search option and from the appropriate form, enter the name of the business function to search.
<b>Business Views</b>	Select a search option and from the appropriate form, enter the name of the business view to search.
<b>Data Structures</b>	Select a search option and from the appropriate form, enter the name of the data structure to search.
<b>Tables</b>	Select a search option and from the appropriate form, enter the name of the table to search.
<b>Forms</b>	Select a search option and from the appropriate form, enter the name of the form to search.

## Searching for Event Rules

Access the ER Search form.

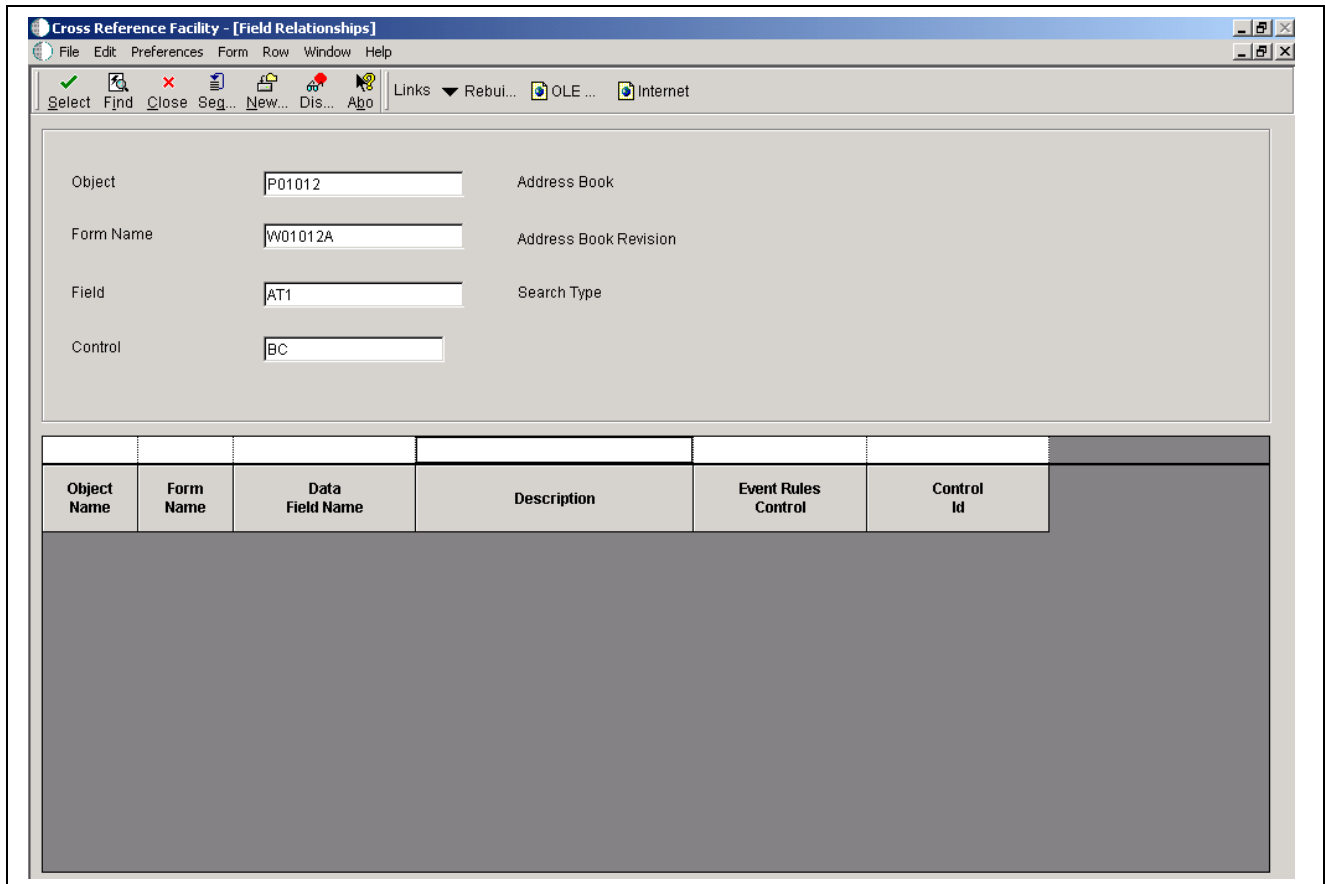


ER Search form

- Object Name**                      The name of the application that you want to search for Event Rules. For example, if you are searching for how Search Type is used in the address book application event rules, enter P01012 in this field.
- Form Name**                        The name of the form that you want to search for Event Rules. For example, if you are searching for how Search Type is used in the Address Book Revisions form, enter W01012A in this field.
- Data Field Name**                The alias of the data item that you want to searching for. For example, if you are searching for how Search Type is used in the Address Book Revisions form, enter the data dictionary item name for Search Type which is *ATI*.

## Viewing Field Relationships

Access the Field Relationships form.



Field Relationships form

- Object** The name of the application that you want to locate field relationships for.
- Form Name** The name of the form that you want to locate field relationships for.
- Field** The alias of the data dictionary item name that you want to locate field relationships for.
- Control** The type of object that you are search for:
  - *BC* - Business View Field.
  - *PO* - Processing Option.
  - *FI* - Form Interconnect.
  - *VAR* - Variable.
  - *FC* - Form Control.
  - *FCW* - Form Control Work Field.
  - *GC* - Grid Column.
  - *GCW* - Grid Column Work Field.

## Rebuilding Cross Reference Information

Access the Cross Reference form.

1. Select Rebuild Relationships from the Form menu.
2. Select the objects that you want to rebuild.

---

**Note.** The rebuild process can take several minutes.

---



## **PART 3**

# **Designing Business Views**

**Chapter 7**  
**Understanding Business Views**

**Chapter 8**  
**Designing Business Views**



# CHAPTER 7

## Understanding Business Views

This chapter discusses business views.

---

### Business Views

A business view is a selection of data items from one or more tables. After you create a table, use Business View Design to create a business view that contains only the data items that are required for the application. PeopleSoft EnterpriseOne uses the business view that you create to generate the appropriate SQL statements necessary to retrieve data from the database. After you define a business view, you can attach it to a form that updates data in an interactive application or to a report that displays data. Because you select only those data items that an application requires, business view increase performance by moving less data across the network.

Business views are required for creating applications and generating reports; they have these characteristics:

- Link a PeopleSoft EnterpriseOne application to one or more tables.
- Contain all or a subset of data items from one or more tables.
- If more than one table is used, table joins are used to join the tables on common fields.

Business views are also a building block for text search indices, which enable full-text searching of data. To make a business view available for full-text searching, select the Text Search option on the properties form. Do not select this option if you do not plan to use the business view for full-text searching; doing so can negatively affect performance.

### Table Joins

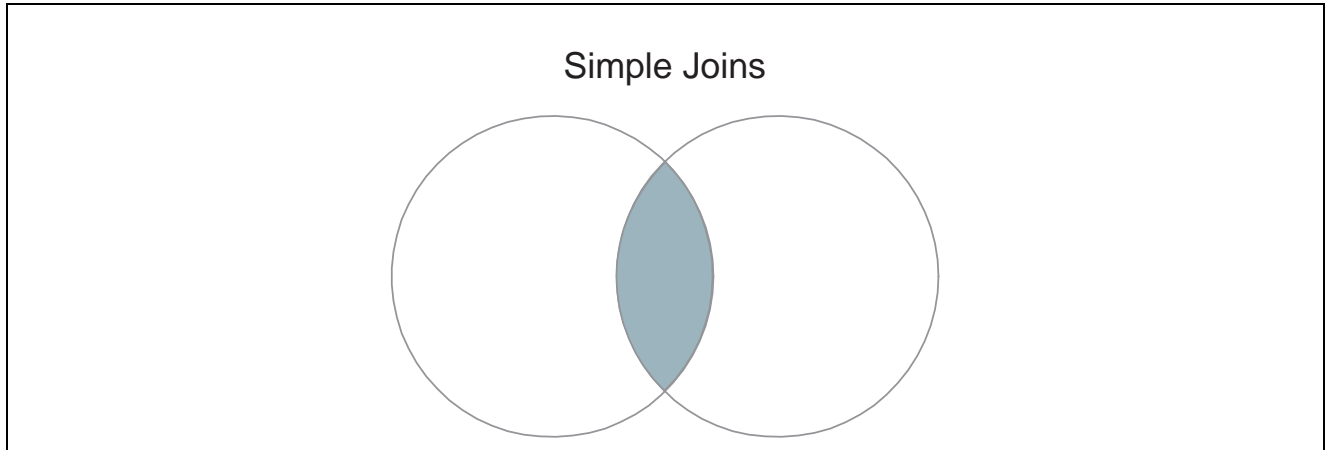
Use the table join feature to join multiple tables in a business view. Joining tables enables you to combine fields from different tables for each record of the primary table. The join is performed using fields common to the tables. The joined fields are defined to satisfy a join condition, such as when the records, or rows, have the same value in the key fields. The primary table is the table where you initiate the join (usually the table on the left in Table Design Aid) and the secondary table is the table where you conclude the join (usually the table on the right in Table Design Aid). Several types of joins exist, including:

- Simple joins, also known as inner joins.
  - Includes only rows that match both the primary and secondary tables.
- Right outer joins.
  - Includes rows common to both the primary and secondary tables, and unmatched rows from the secondary table.
- Left outer joins.
  - Includes rows common to both the primary and secondary tables, and unmatched rows from the primary table.

- SQL 92 left outer joins.

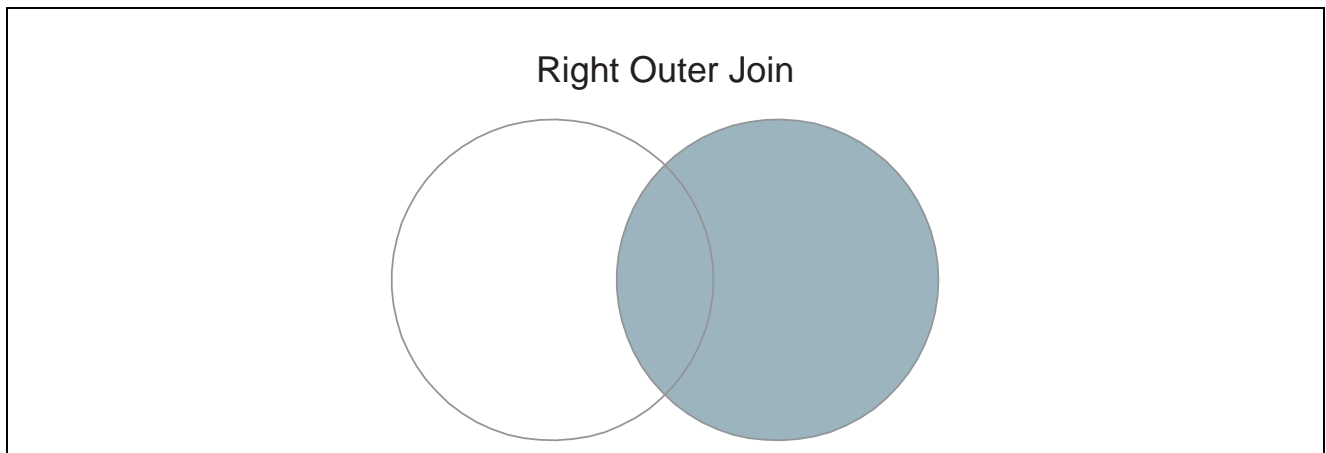
Includes rows common to both the primary and secondary tables, unmatched rows from the primary table, and any rows with null values from the secondary table - regardless of any Where clause against the fields from the secondary table.

This graphic illustrates a simple table join:



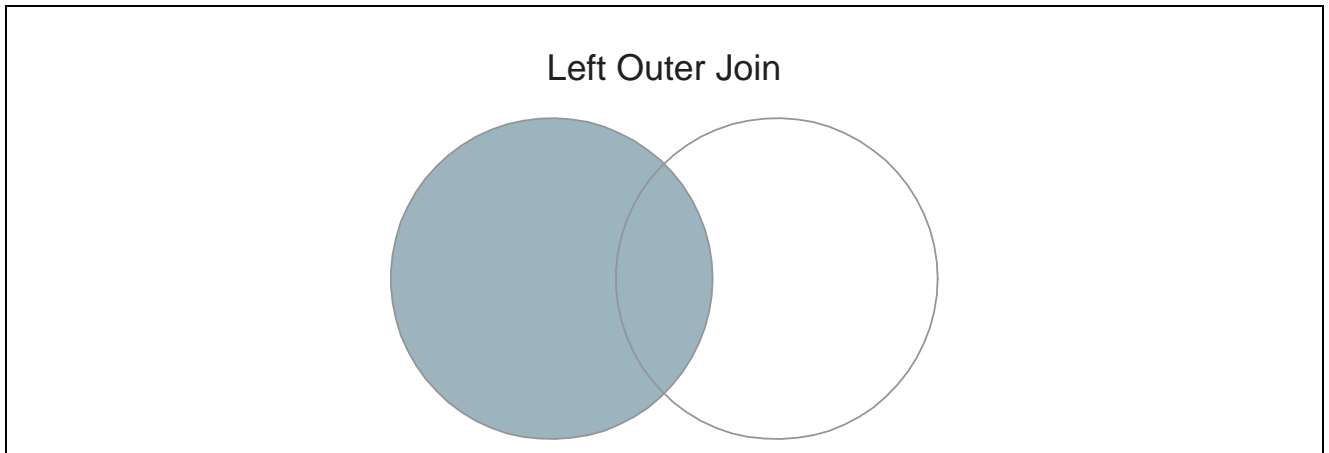
Simple Join

This graphic illustrates a right outer table join:



Right Outer Join

This graphic illustrates a left outer table join:



Left Outer Join

## Table Union

A table union joins entire tables. The system first checks for rows from the primary table, and then for rows with corresponding columns from the secondary table. If the rows from the two tables contain identical data, then only one of the records is retrieved in the union. Unions include rows from the primary table and corresponding columns from the secondary table.

This diagram illustrates a union:

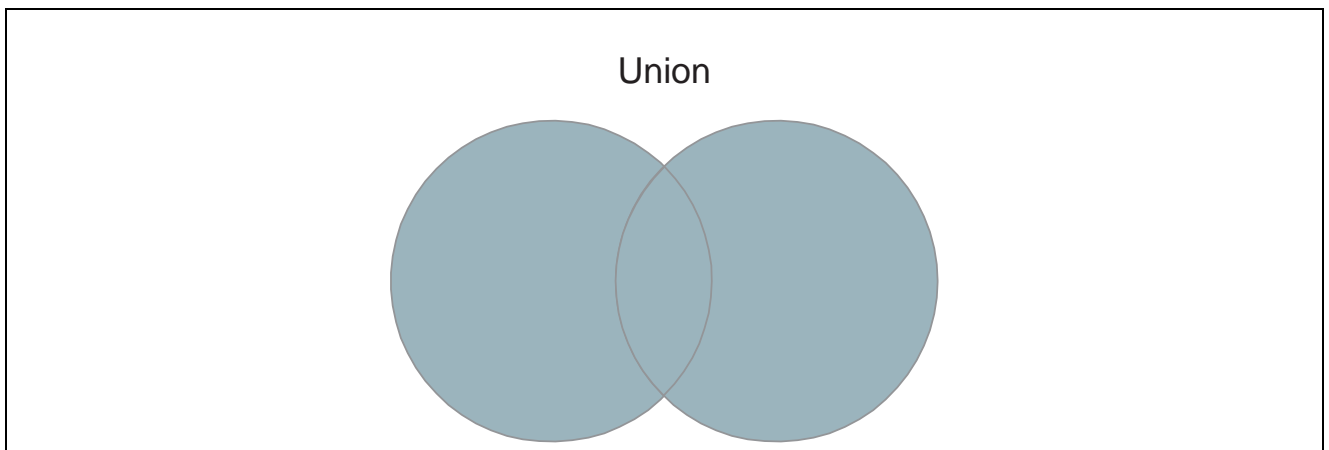


Table Union

## Select Distinct

If a business view includes the primary key fields of the primary table, every row of the business view query is unique. The primary key field has a different value in each row, or record, of the primary table. If the business view does not contain all primary key fields of the primary table, then duplicate rows can occur during the business view query. You can eliminate the duplicate rows in the output by using the Select Distinct feature when designing the business view.

## Primary Key Fields

The fields that are included in the primary index of a table are displayed as key fields in Business View Design. These fields include a key icon next to the field name. The primary key fields of a table are always included in the business view. Business views then carry information from the table to an application. To carry forward additional information other than the primary key fields, select additional fields to include in the business view.

## CHAPTER 8

# Designing Business Views

This chapter provides an overview of business view design, business view naming conventions, Business View Design Aid, and discusses how to create business views.

---

## Understanding Business View Design

Before you begin designing a business view, consider the purpose of the application that will use the business view and the data items required. You can then identify the tables that contain the required data items. Adding a new business view does not affect performance; however, using an existing business view that contains many columns that are not required might negatively affect performance.

Business views usually contain more fields than are used on the form, in the grid, or in the batch application. The unused fields are related to the fields that are used. If requirements change, these fields can be added to the application without redesign of the business view.

You can modify business views to reflect changes in business requirements. It is easy to add fields to existing business views. Deleting fields from business views is more complicated. If you need to delete fields from a business view, ensure that those fields are not currently in use in an interactive or batch application.

You can use different business views for each form type included in an interactive application. Typically, search-and-select forms include the minimum number of fields required to keep them at a nominal size. They should include only the fields necessary for filtering searches and the associated output fields, such as descriptions.

Find/browse and parent/child forms include more fields and are typically larger than search and select forms. You should still limit the number of fields on find/browse and parent/child forms to include only those fields required for filtering and displaying the associated output fields.

Input-capable forms are usually large and include all of the fields from the table. They include all of the fields necessary to add or update a record, including audit information.

---

## Understanding Business View Naming Conventions

Use these guidelines when naming a business view:

The object name for a business view can be a maximum of 8 characters and should be formatted as:

*VzzzzzzA*, where:

*V* = Business view.

*zzzzzz* = The characters that represent the primary table.

*A* = A letter that designates the view.

For example, V0101A is the first business view created over the F0101 table, V0101B is the second business view created over the same table, V0101C is the third business view created.

## External Developer Considerations for Naming Business Views

The term *external development* refers to applications created by developers who are not employed by PeopleSoft, such as consultants who create custom applications for clients. To prevent interference between PeopleSoft EnterpriseOne and non-PeopleSoft EnterpriseOne objects, you must use caution when naming an external business view. When you create business views for an application created by external developers, format the business view name as:

*V*ssss9999, where:

*V* = Business view.

ssss = The system code for the enterprise.

9999 = A unique next number or character pattern within the enterprise.

Provide a business view description with a maximum of 60 characters. It should reflect the application description followed by the form type, for example, Item Master Browse and Item Master Revisions.

Primary key fields should remain in the business view and should not be reorganized. If you try to remove a primary key field, you receive an error message.

---

**Note.** At least one business view for each table should include all columns from the table. Only one business view is allowed for each form type, except for a header/detail form. You can use two business views on header/detail forms, one for the header portion of the form and one for the detail portion.

---

## Naming Joined Business Views

To format the description of a joined business view, include the names of the tables being joined, separated by a forward slash (/). Place the primary table first.

For example, a join view is created by joining the F4101 (Item Master) table and the F4102 (Item Branch) table, and the F4101 table is the primary table, then the view should be named:

Item Master and Branch F4101/F4102

---

## Understanding Business View Design Aid

Business View Design Aid presents these forms within a single window:

- Table Joins.

Defines the tables over which you create the business view. The Table Joins form displays the tables that you select, along with the columns included in the tables. A key icon appears next to the primary key fields. The primary key fields are fields that are included in the primary index of the table. The primary table is where an application begins a search.

- Available Tables.

Used to locate tables for moving to the Table Joins form.

- **Selected Columns.**  
Lists the data items selected from the table to include in the business view.
- **Object Properties.**  
Displays the properties of a data item selected in the Selected Columns form.
- **Available GT (Generic Text) Data Structures**  
Used to locate generic text data structures for moving to the Selected GT (Generic Text) Data Structures form. GT data structures are used for text search business views only.
- **Selected GT (Generic Text) Data Structures**  
Lists the generic text data structures selected to include in the business view. GT data structures are used for text search business views only.

Note these points when working with business views:

- If you delete a data item from a business view and that data item is used in an application, you receive an error when you attempt to run the application.  
If this occurs, you must open the application and delete the data item from the application.
- When you delete an entire table from a business view, you cannot run any of the applications that use the business view.  
If this occurs, you must open the application and delete all items that reside in the deleted table or attach a different business view to the application and connect all of the controls.
- If you delete a business view, any forms that use the business view will fail.  
If this occurs, you must associate the forms to a new business view and connect all of the controls.

## Selecting Tables for Business Views

You must select tables from which to create business views. The tables you select must include the fields required to meet the business purpose. Although you can create one large business view to retrieve and update the columns required by the application, performance is negatively impacted. Whenever possible, consider using table joins rather than creating one large business view that contains many data items.

---

**Note.** To ensure maximum performance in applications, use these guidelines for the number of tables joined in a business view:

Five tables if all joins are simple joins.

Three tables if any of the joins is an outer join or in the event of a union.

---

### See Also

[Chapter 8, “Designing Business Views,” Creating Table Joins, page 57](#)

## Selecting Data Items for Business Views

After you select tables for the business view, you must indicate the primary table. You can then select the required data items to include in the business view. All data items in the selected tables are available for the business view.

Select the data items required by the interactive or batch application. When you create an application, you do not have to use every item in the business view. Balance between keeping the business view small for maximum performance and including enough fields to allow for future business requirements.

If the required data item appears in multiple tables, you typically want to select the data item from the primary table. Selecting the same data item from multiple tables causes the data item to appear multiple times in the business view. Each data item has its own table reference to identify its origin.

---

**Note.** If you include multiple tables in a joined business view, the primary key fields are automatically selected from the tables. You cannot remove the primary key fields from the business view.

---

If you include multiple tables in the business view, you must join the tables using a table join. You typically want to join tables on common key fields. It might be necessary to also join on additional fields. You should join on as many fields as necessary to ensure that the data is fetched properly for each record. The joins must be performed on fields of the same field type.

---

**Important!** To ensure maximum system performance, do not include more than 256 columns in business views.

---

## See Also

[Chapter 8, “Designing Business Views,” Creating Table Joins, page 57](#)

## Using Select Distinct

If a business view includes the primary key fields of the primary table, every row of the business view query is unique. If the business view does not contain all primary key fields of the primary table, then duplicate rows can occur during the business view query.

For example, Journal Entry is unique by line number and document number. However, each document can contain multiple lines. If you need to display only one record per document, you can use Select Distinct to fetch only the first occurrence of the document number, not all of the detail lines within the document.

Any business views with a primary table that contains any of the columns in this table, which are used for currency support and security, might cause the Select Distinct feature to output duplicate values:

Column	Description
CO	Company
CRCD	Currency Code - From
CRDC	Currency Code - To
CRCX	Currency Code - Denominated In
CRCA	Currency Code - A/B Amounts
LT	Ledger Type
AID	Account ID
MCU	Business Unit

Column	Description
KCOO	Order Company (Company Code)
EMCU	Business Unit Header
MMCU	Branch
AN8	Address Number

### Example: Select Distinct Feature

The business view used for this example, Event Detail Business View (V98EVDTL), uses the primary index of the primary table, F89EVDTL (Event Detail File), by default. The primary index of a PeopleSoft EnterpriseOne table must be unique. A unique primary index ensures that the system does not return duplicate values when the business view query is generated. However, Business View Design enables you to use any other index of the primary table when you process the business view.

These steps illustrate how the Select Distinct feature works:

1. In Business View Design Aid, select the primary table.
2. From the Table menu, select Change Index to change the primary table index.

Change Index is available only for the primary table.

The system displays a warning, indicating that the selected column list will be changed.

3. Click Yes to continue.

The Available Indices form appears. The first edit field on the form displays the current index of the table used by the business view. The default is the primary index.

4. For this example, select *Key by Formtyp, Evtype, Obj* from Available Indices, and click OK.

The Table Joins list and Selected Columns list reflect the keys of the new index.

5. Save the changes and exit Business View Design Aid.

Now, if you run an application that uses the V98EVDTL business view with Select Distinct disabled and the changed business view index (Key by Formtyp, Evtype, Obj), the generated SQL statement is:

```
SELECT EDOBJTYPE, EDEVTYPE, EDFORMTYPE FROM PVC. F98EVDTL
```

Using this example, you might now have 281 rows of data from table F98EVDTL.

6. Reopen the V98EVDTL business view.
7. From the File menu, click Select Distinct.
8. Select Change Index to select the *Key by Formtyp, Evtype, Obj* index from Available Indices and then click OK.
9. Save the business view and exit Business View Design Aid.

You might need to exit the software and sign back in. The software stores the business view in cache memory, even though you changed the business view, the previous business view runs until it is cleared from cache.

Generate and rerun the same application using the V98EVDTL business view with Select Distinct activated. The generated SQL statement is now:

```
SELECT DISTINCT EDOBJTYPE, EDEVTYPE, EDFORMTYPE FROM PVC. F98EVDTL
```

\_Continue2

Using this example, you might now have only 53 rows of data from table F98EVDTL.

## Creating Table Joins

Create table joins in business views to access the data from multiple tables in a single application.

You typically use joins for forms that are not input-capable, such as find/browse forms, and reports. You do not usually use joins for forms that update and add to the database, because the relationship between the records must be precise. If you must use a join for an input-capable form, only use a join where the relationship between the two tables is simple.

If a business view uses multiple tables, link the tables by establishing joins between columns in those tables. The links define how rows from one table correspond to rows in another table.

When joining a column in one table to a column in another table, both columns must be of the same data type. You can use the Object Properties form to view attributes for a column to determine whether you can use it in a join. The Object Properties form displays the attributes of the data item that you have highlighted on the Selected Columns form.

Review each table and determine how the data in one table is related to the data in the other tables. You might need to add columns or build new indices in a table, or even create new tables. If you build new indices, consider your business needs carefully before you do so.

---

**Note.** To ensure maximum performance in applications, use these guidelines for the number of tables joined in a business view:

Five tables if all joins are simple joins.

Three tables if any of the joins is an outer join or in the event of a table union.

---

To join columns you must draw a line, using the mouse, from a column in the primary table to the associated column in a secondary table. When you click the line that you drew, you can define the join by selecting a join type and an operator from the Join menu.

The default join type is simple and the default operator is equal. Other available operators are:

- Equal (=)
- Less than (<)
- Greater than (>)
- Less than or equal (<=)
- Greater than or equal (>=)

## Creating Table Unions

Unions are used to pull rows from tables that have the same structure. Unions pull rows that exist in both tables. The table union option is only available if the rows in one table are also included in the other table.

## Creating Business Views

This section discusses how to:

- Add business views.
- Select tables for business views.
- Select data items for business views.
- Use Select Distinct.
- Create table joins.
- Create table unions.

## Forms Used to Create Business Views

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	EnterpriseOne Life Cycle Tools, Application Development (GH902), Object Management, Object Management Workbench	Select and view objects in projects and access the PeopleSoft EnterpriseOne design tools.
Add EnterpriseOne Object to the Project	W98220C	Click the Objects node of a project and click Add on the Object Management Workbench form.	Add a new object to a project.
Add Object	W9861AF	Select Business View and click OK on the Add EnterpriseOne Object to the Project form.	Create a business view.
Business View Design Aid	W9860AL	Complete the object information and click OK on the Add Object form.	Access Business View Design Aid and create a type definition.

## Adding Business Views

Access the Add Object form.

<b>Object Name</b>	A unique name for the business view. Use the recommended naming convention for naming PeopleSoft EnterpriseOne objects.
<b>Description</b>	A meaningful description of the business view.
<b>Product Code</b>	A user defined code (98/SY) value. Use one of the values from the range of values reserved for clients, 55–59.
<b>Product System Code</b>	A user defined code (98/SY) value that represents the PeopleSoft EnterpriseOne system where the data is used. This value is used for reporting and jargon purposes. Example values include: <i>01</i> : Address Book <i>03B</i> : Accounts Receivable

	<i>04: Accounts Payable</i>
	<i>09: General Accounting</i>
	<i>11: Multicurrency</i>
<b>Object Use</b>	A value that indicates the use of the object. For example, the object may be used to create a program, a master file, or a transaction journal. The value 300 represents business views.
<b>Object Type</b>	An abbreviation that identifies the type of object being created. The object type for a business view is BSVW. This is a display field that is populated by the system based on the type of object that you selected on the Add Object form.
<b>Text Search</b>	Indicates that the business view is used by the runtime engine during a text search. When designing the business view you must also add at least one GT (general text) data structure to the business view so that the runtime engine can use it in text searches.

## Selecting Tables for Business Views

Access the Business View Design Aid form.

1. On the Design Tools tab, click Start the Business View Design Aid.
2. On the Available Tables form, use the QBE line to search for an appropriate table
3. Select one or more tables and drag them to the Table Joins form.

This form is called Table Joins regardless of whether you are joining multiple tables or working on a single table.

4. If more than one table is selected, double-click the title bar of the appropriate table to designate it as the primary table.

If the business view contains multiple tables, the system automatically designates the first table added as the primary table. A crown icon appears in the upper-left corner of the primary table. If a business view contains only one table, that table is the primary table by default.

---

**Note.** To delete a table from a business view, select the table and select Delete from the Table menu. You can also right-click the table and select Delete from the pop-up menu.

---

## Selecting Data Items for Business Views

Access the Business View Design Aid form.

1. On the Table Joins form, double-click the data items to include in the business view.  
Selected data items appear with a check mark on the Table Joins form. As you select each data item, the system adds it to the Selected Columns form.
2. To remove data items from a business view, double-click the data item either on the Table Joins form or on the Selected Columns form.

## Using Select Distinct

Access the Business View Design Aid form.

1. Select the primary table of the business view.

2. From the Table menu, select Distinct Mode.
3. From the Table menu, select Change Index.

This option changes the index of the primary table to a non-unique index.

## Creating Table Joins

Access the Business View Design Aid form.

1. On the Tables Joins form, click and draw a line that connects a column in the primary table to a column in a secondary table.

Although the column names do not have to be the same, the attributes for Data Type and Decimals must be identical before you can create table joins between two columns. To determine whether data items are candidates for a join, click a data item on the Table Joins form and view the data item attributes displayed on the Object Properties form.

2. Click the line that joins the two columns.

Both fields are highlighted.

3. From the Join menu, select Types and then select an appropriate join type.

Simple is the default join type.

4. From the Join menu, select Operators and then select an appropriate operator.

5. To delete a join, click the line that connects the two fields and then select Delete from the Join menu.

You can also right-click the join and select Delete from the pop-up menu.

## Creating Table Unions

Access the Business View Design Aid form.

1. From the Table menu, select Union Mode.

You can also click the Union Mode button on the toolbar.

The Union Mode option is available only if all columns in one table also reside in the other table.

2. Select the tables for which you want to create a table union.



## **PART 4**

# Converting Tables

**Chapter 9**  
**Understanding Table Conversion**

**Chapter 10**  
**Setting Up Table Conversions**

**Chapter 11**  
**Running Table Conversions**

**Chapter 12**  
**Preparing Foreign Tables for Table Conversion**



## CHAPTER 9

# Understanding Table Conversion

This chapter discusses:

- Table conversions.
- Types of tables to use.
- Business views in table conversions.
- Text files in table conversions.
- Sort and selection criteria in table conversions.
- Input and output environments.

---

## Table Conversions

Table conversion is a type of batch process that enables you to rapidly manipulate the data in tables. The Table Conversion tool uses these conversion types to enable you to manipulate data in a variety of ways:

- Data Conversion

Enables you to transfer or copy data from an input table or business view into output tables using the logic necessary to perform the transfer. You can also use Data Conversion to update records in a table or business view.

- Data Copy

Enables you to copy tables from one data source or environment to another data source or environment when the tables are identical.

- Data Copy with Table Input

Enables you to copy tables based on information from an input table. For example, the input table might provide information about which tables are copied, where they are copied, and so on.

- Batch Delete

Enables you to delete records from a table or business view.

The Table Conversion tool uses PeopleSoft EnterpriseOne tables, business views, and text files. The tool can also use non-PeopleSoft EnterpriseOne tables, also referred to as *foreign tables*, if they reside in a database supported by PeopleSoft EnterpriseOne software; such as Oracle, Access, iSeries, or SQL Server.

You first set up a table conversion, then you review it, and then you run it. The conversion can be saved and run multiple times. You can test the conversion by running it in proof mode.

Like batch applications, table conversions include a template and versions. You can override some of the properties within a version at runtime.

Table Conversion Design enables you to access any available environment for both input and output. The environments that you select determine which tables and business views are available for the conversion and where the tables reside. The environments that you select also determine the specifications, or descriptions, of tables and business views.

---

## Types of Tables to Use

You can use these table types in table conversions:

- PeopleSoft EnterpriseOne tables.

These tables exist in the PeopleSoft EnterpriseOne database and can be accessed from Object Management Workbench. You can design and modify them using Table Design Aid. At design time, only the specifications are needed to reference the table. At runtime, the table must be generated. An instance of the table must occur in a particular database.

- Foreign tables.

These tables do not have a PeopleSoft EnterpriseOne definition but must reside in a database supported by PeopleSoft EnterpriseOne. You must set up a data source and environment in PeopleSoft EnterpriseOne to point to the location of foreign tables.

### See Also

[Chapter 12, “Preparing Foreign Tables for Table Conversion,” page 107](#)

---

## Business Views in Table Conversions

To transfer data from multiple tables to a single table or from multiple tables to multiple tables, you must establish a relationship among the input tables by creating a business view.

A business view defines the relationship between two or more tables and joins the data into a single view. You can use PeopleSoft EnterpriseOne business views for input to the table conversion only, not for output.

The system does not provide direct support for joining foreign tables. To use multiple foreign tables as input to a conversion, you must first define them through PeopleSoft EnterpriseOne and then create a business view for them.

---

## Text Files in Table Conversions

You can import directly from or export directly to a text file. When you convert a text file, it is stored with a single, long text field. User defined formats are stored the same for a text file as for any table. With a text file conversion, the table name includes the path and the file name. If you do not specify the path with the file name, the default path is used.

---

## Sort and Selection Criteria in Table Conversions

You can specify sort criteria in a table conversion. Sorting is used to process input rows in a sequence that groups related records together. The sort and selection features simplify the process of writing records to multiple tables in a typical one-to-many conversion. For example, if you have a single table of customer information, you can sort the data by area code and split the table into individual tables for each area code. Similarly, you can specify selection criteria for the input table to convert only a subset of that table.

Table Conversion Design also enables you to add logic to determine when a change occurs to the value in a field.

---

## Input and Output Environments

An environment consists of a path code and Object Configuration Manager (OCM) mappings that indicate where PeopleSoft EnterpriseOne objects reside. Table Conversion Design enables you to specify an input and output environment, which is used to locate input and output tables. To locate foreign tables, the Table Conversion tool uses the default OCM mapping for tables.

The path code of the environment is used to locate specification files for the environment. This path code is usually a subdirectory of the PeopleSoft EnterpriseOne directory on a workstation. To reference PeopleSoft EnterpriseOne tables in an environment, the full path code must exist on the machines where the conversion is created and run. Foreign tables can be referenced even if the path code does not exist.

The Table Conversion tool uses these three environments when it processes a table conversion:

- The environment that you are signed into.  
Determines where the table conversion specifications are stored.
- The environment where the input tables reside.
- The environment where the output tables reside.

You can select one environment for the input tables or business views and one for the converted output tables. The three environments can be the same, or they can be different.

When you use PeopleSoft EnterpriseOne tables or business views in a table conversion, the environment provides the details of each table or business view, such as column names, data types, and descriptions. Because this information comes from the PeopleSoft EnterpriseOne specification tables, the system table or business view does not have to exist in the database before you can design a table conversion; however, it must be created before the conversion is run. If you use a foreign table as input, you need to create it before you design the table conversion. The Table Conversion tool obtains its information about the table directly from the database where the foreign table resides. The environment also provides a default path for text files.

**See Also**

Part 1, “Designing Tables in PeopleSoft EnterpriseOne,” page 1

Chapter 12, “Preparing Foreign Tables for Table Conversion,” page 107

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, “Setting Up Path Codes”*

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, “Setting Up Environments”*

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, “Setting Up Data Sources”*

## CHAPTER 10

# Setting Up Table Conversions

This chapter provides overviews of table conversion setup, and the table conversion process flow, and discusses how to:

- Start the table conversion director.
- Convert data.
- Define file formats for table conversion input or output.
- Copy data.
- Copy data with table input.
- Delete groups of records.
- Use event rules in table conversions.

---

## Understanding Table Conversion Setup

Table Conversion Design Aid uses a director to guide you through the steps of setting up table conversions. Using the director, you can design table conversions for converting data, copying tables between locations, and deleting records within tables.

You can also define user defined formats to use in the table conversion when you select the input or output tables. User defined formats are flat file tables that store data as a continuous string of information, such as bank tapes.

After you set up a table conversion, you can save it and run it multiple times.

### See Also

[Chapter 10, “Setting Up Table Conversions,” Understanding User Defined Formats, page 77](#)

---

## Understanding the Table Conversion Process Flow

When you run a table conversion, the system triggers events that are similar to the events that are triggered when a report or application is run. These events are specific to the table conversion that you defined and they provide pauses in the execution where you can attach logic.

In general, the event flow is the same for all table conversion types, Data Copy, Data Copy with Table Input, and Batch Delete, because these conversion types are all subsets of a data conversion. For example, the Data Copy conversion type does not include input and output tables; all actions are accomplished through the Process Begin event. The Data Copy with Table Input and Batch Delete conversion types do not include output tables; all actions are accomplished through the Process Begin, Process End, and Row Fetched events. This flexibility enables you to use table conversion types within other conversion types, if necessary.

Depending on the type of conversion that you define, some events might not be triggered. This diagram illustrates all events that can be triggered in a table conversion; events are indicated by a yes-or-no decision:

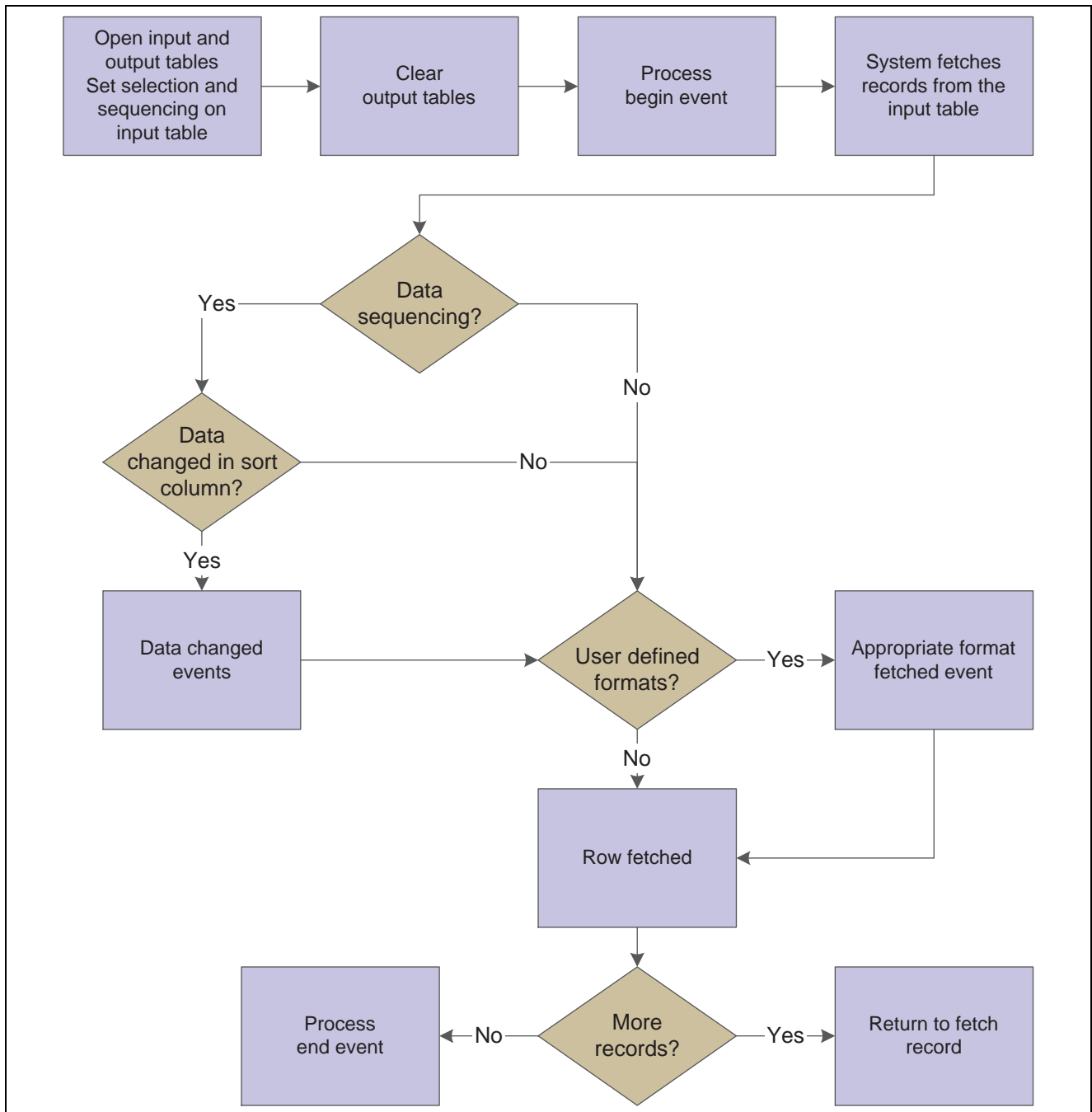


Table conversion event flow.

Events in table conversions occur in this order:

1. Process Begin.

Before fetching records from the input table, the system invokes the Process Begin event. At this point, you can attach logic that needs to run only once at the beginning of a table conversion, or any value that does not change for each record. This event is useful for mapping output fields that do not change for each record.

2. Data Changed.

When you use data sequencing, the system invokes a Data Changed event for any sequenced field that changes. Data Changed events are not cascaded or hierarchical. For example, you can attach an event rule to this event to total a field or group of values.

3. Format Fetched.

When you use user defined formats (also known as flat files) in the input table, the system invokes a Format Fetched event for each record fetched from the input table. If you use multiple user defined formats in a table conversion, the Format Fetched event is called for the particular format that is found in the record.

4. Row Fetched.

An input table invokes a Row Fetched event after each row is fetched from the input table.

5. Process End

After the processing of all records is complete, the system invokes the Process End event. Attach event rules to the Process End event when you want the system to process logic after all input records have been read; for example, to write a total record to an output table or to write a record to a log file to record the status of the table conversion.

---

## Starting the Table Conversion Director

This section provides an overview of the navigation assistant, lists the prerequisites, and discusses how to:

- Add new table conversions.
- Start the Table Conversion director.

## Understanding the Navigation Assistant

The Table Conversion director includes a navigation assistant. The navigation assistant provides a visual indicator of your progress in designing the table conversion. Click a step on the navigation assistant to move to another step in the process.

## Prerequisites

Before you use the table conversion process, perform these tasks.

- If you are importing data from foreign tables, set up a data source and environment for those tables.  
See [Chapter 12, “Preparing Foreign Tables for Table Conversion,” page 107](#).
- If you are converting data from multiple tables, create a joined business view for the tables.  
See [Chapter 7, “Understanding Business Views,” Table Joins, page 45](#).
- If you want the data items within a table to validate against the data dictionary as part of the conversion process, create a business function to perform the validation.

The Table Conversion tool does not provide automatic data dictionary validation for inputs or outputs.

See *PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: APIs and Business Functions*, “Using Business Functions”.

## Forms Used to Add New Table Conversions

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	EnterpriseOne Life Cycle Tools, Application Development (GH902), Object Management, Object Management Workbench	Select and view objects in projects and access the PeopleSoft EnterpriseOne design tools.
Add EnterpriseOne Object to the Project	W98220C	Click the Objects node of a project and click Add. on the Object Management Workbench form.	Add a new object to a project.
Add Object	W9861AF	Select Batch Application and click OK on the Add EnterpriseOne Object to the Project form.	Create a batch application.
Batch Application Design	W9860AL	Complete the object information and click OK on the Add Object form.	Access the Table Conversion director.

## Adding New Table Conversions

Access the Add Object form.

<b>Object Name</b>	A unique name for the table conversion. Use the recommended naming convention for PeopleSoft EnterpriseOne objects.
<b>Description</b>	A meaningful description of the table conversion.
<b>Product Code</b>	A user defined code (98/SY) value. Use one of the values from the range of values reserved for clients, 55–59.
<b>Product System Code</b>	A user defined code (98/SY) value that represents the PeopleSoft EnterpriseOne system where the data is used. This value is used for reporting and jargon purposes. Example values include: <i>01: Address Book</i> <i>03B: Accounts Receivable</i> <i>04: Accounts Payable</i> <i>09: General Accounting</i> <i>11: Multicurrency</i>
<b>Object Use</b>	A value that indicates the use of the object. For example, the object may be used to create a program, a master file, or a transaction journal. A 132 value represents a data file conversion.

<b>Object Type</b>	An abbreviation that identifies the type of object being created. The objects type for a table conversion is UBE. This is a display field that is populated by the system based on the type of object that you selected on the Add Object form.
<b>Table Conversion</b>	Enables the Table Conversion Design tool. If you do not select this option, the Report Design Aid tool is launched by default.

## Starting the Table Conversion Director

Access the Table Conversion Design form.

1. Select the Design Tools tab, and click Start Table Conversion Design Aid.  
The system displays the Introduction form for the Table Conversion director.
2. Proceed to the steps for the type of conversion that you want to perform.

### See Also

[Chapter 10, “Setting Up Table Conversions,” Converting Data, page 69](#)

[Chapter 10, “Setting Up Table Conversions,” Copying Data, page 82](#)

[Chapter 10, “Setting Up Table Conversions,” Copying Data with Table Input, page 84](#)

[Chapter 10, “Setting Up Table Conversions,” Deleting Groups of Records, page 86](#)

---

## Converting Data

This section provides an overview of data conversions and discusses how to:

- Define external data for table conversions.
- Define the input and output environments for table conversions.
- Define table conversion input.
- Define table conversion output.
- Map inputs to outputs.
- Specify table conversion logging options.
- Review table conversion results.

## Understanding Data Conversion

Use the Data Conversion option on the Table Conversion director to move data into tables from:

- A single table.
- Multiple tables defined in a business view.
- A single text file.

## Data Conversion Example

This data conversion example copies employee records from the F0101 table in the Login Environment to the F0101 in the PY811 environment.

1. On the Introduction form of the Table Conversion director, select Data Conversion, and click Next.
2. On the External Data form, accept the default selections, and click Next.
3. On the Select Environments form, select `<LOGIN ENV>` as the input and `PY811` as the output environment, and click Next.
4. On the Select Input form, on the Table tab, drag the Address Book Master table to the Description column and click Next.
5. On the Sequencing form, accept the default selection, and click Next.
6. On the Data Selection form, define the data selection as *Where ATI (Search Type) (F0101) is equal to E* to select current employees, and click Next.
7. On the Select Outputs form, on the Tables tab, drag the Address Book Master table to the Description column and click Next.
8. On the Table Options form, clear the Run Currency Triggers option since the Address Book Master file does not contain currency information, and click Next.
9. On the Mapping form, select the Row Fetched event since you are converting data from similar files, click Map Same and click Next.

The system automatically maps data elements with the same aliases.

10. On the Logging Options form, select Log All Errors and Log Outputs and click Next.

These options enable you to review the conversion after it is complete

11. On the Finish form, select No, I will create a version of this table conversion later and click Finish.
12. On the Table Conversion form, click Save before exiting.

## Defining External Data for Table Conversions

Access the Table Conversion Director.

1. Select Data Conversion and click Next.
2. On the External Data form, click Select to attach a predefined processing option template to the table conversion.
3. On the Select Processing Option Template form, select an appropriate processing option template and click OK.
4. On the External Data form, click Define to create a data structure.
5. On the Report Data Structure form, define the data structure to attach to the table conversion, and click OK.

Data structures contain a list of parameters that can be used to pass data into the table conversion when called through a report interconnect.

6. Click Next.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Data Structure Design, "Understanding Data Structures"*

## Defining Input and Output Environments for Table Conversions

Click Next on the Introduction form of the Table Conversion director to access the Select Environments form.

1. Select the input and output environments to use.

---

**Note.** Select <LOGIN ENV> if you are creating table conversions for a user who might not have the same environments that you have. This option ensures that the table conversion uses the environment that the user logs into.

---

2. When creating a table conversion to be run in a different environment from the one where you create it, and the <LOGIN ENV> is not appropriate for the type of conversion that you are creating, select either Force Version to Override Input Environment or Force Version to Override Output Environment and click Next.

For example, if you create a conversion to be used by a client who does not have the environments that you have, use these options. When the conversion is invoked at the client site, the table conversion will not run until the user selects the appropriate environments in which to run it.

## Defining Table Conversion Input

Click Next on the Select Environments form to access the Select Input form.

1. Select the appropriate tab based on whether the input is a table, a business view, a foreign table, or a text file.
2. Use the QBE line to search for the appropriate table, business view, or foreign table.

You can select only one table or one business view per table conversion. If the input consists of multiple tables, you must create a joined business view over the tables.

3. Enter the name of the text file on the Text File tab or use the Browse button to locate the file.

If you would like to rename the selected file, enter the new file name, and click Rename.

For the iSeries, input text files are stored in the Integrated File System (IFS). Enter the path to the IFS before the file name.

If you are using a text file or if you need to define a format for a table or business view, select the User Defined Format option.

---

**Note.** Text files are the only type of file that you can rename.

---

4. If the input is a table or a business view, drag it to the Description column.

If you change a table or business view, the system warns you that deleting tables removes all mappings from the table conversion.

5. To delete an input file, select it and press DELETE.
6. Click Next.
7. On the Sequencing form, click Data Sequencing to define data sequencing for a table or business view.

If you specify a text file for input, you cannot define data sequencing or data selection for that file.

---

**Note.** When you define data sequencing, you create new events that are available to you in the Mapping section of the director. One new event is created for each sequence column that you define. The event is called *XXXX* Data Changed, where *XXXX* is the column alias; for example, *ALPH* Data Changed. Each time the value in one of these columns changes from its previous value, the Data Changed event is invoked for that column. This event is similar to a level break in report writing, except that the Data Changed events are not related to each other. Invoking one does not invoke the others.

---

8. Click Next.
9. On the Data Selection form, click in the Operator column and select an appropriate operator.

On the Data Selection form, *Where* is the default value in the Operator column for the first set of criteria. For subsequent statements, *And* and *Or* become the available values for the Operator column.

10. Click in the Left Operand column and select from a list of available objects.

---

**Note.** You can define selection criteria for database table columns only. User defined format columns are not available because they do not exist in the database.

---

11. Click in the Comparison column and select one of these comparison operators:

- is equal to
- is equal to or empty
- is greater than
- is greater than or equal to
- is less than
- is less than or equal to
- is not equal to

12. Click in the Right Operand column and select from a list of available objects, special values, or variables.

The options in this column depend on the selection that you made in the Comparison column. Some of these options might be available:

- Blank  
Enters a blank (space) value.
- Literal  
Enters specific values as described in the next step.
- Null  
Indicates that no value is associated with the field.
- Zero  
Enters a value of 0.
- IC  
Indicates input table columns.
- RI  
Indicates values passed through a report interconnect to this table conversion.

- PO  
Indicates processing option values for this table conversion.
  - SL  
Indicates system literals.
13. If you enter a literal in the Right Operand column, you can enter values on these tabs of the Single value form:
- Single value  
Enter a single value. An example might be a value that indicates a specific company.
  - Range of values  
Enter a range of values. An example might include companies from *00001* to *00060*. When using a range of values, only the *is equal to*, *is equal to or empty* and *is not equal to* operators are valid.
  - List of values  
To enter a list of values, type each value in the field and click Add.  
An example list of values might include several user defined codes for search types, such as *C* for customers, *E* for employees, and *V* for vendors. When using the list of values, only *is equal to*, *is equal to or empty* and *is not equal to* are valid operators.  
To delete a value, select the value and click Delete.
14. To delete a line of criteria on the Data Selection form, select the row header, and click Delete.
15. To change the order of the criteria, select the row header, and click the up or down button.
16. Click Next.

## Defining Table Conversion Output

Click Next on the Select Input form to access the Select Outputs form.

1. Drag the table (or tables) that you want to use as outputs to the Description column, and click Next.
  2. For text file conversions, from the Text File tab, select the file to use as the output and click Use.  
If you would like to rename the selected file, enter the new file name, and click Rename.  
For the iSeries, input text files are stored in the Integrated File System (IFS). Enter the path to the IFS before the file name.  
If you are using a text file or if you need to define a format for a table or business view, select the User Defined Format option.
- 
- Note.** Text files are the only type of file that you can rename.
- 
3. Click Next.
  4. To delete an output, select the row and press DELETE.  
If you are using a text file, or if you need to define a user defined format for a table or business view, click User Defined Format.
  5. Click Next.
  6. On the Table Options form, select from these options:

- Run Currency Triggers

Select this option if a PeopleSoft EnterpriseOne table that is included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option and the source or destination fields are currency fields used in a calculation, you might receive unexpected results.

Do not select the Run Currency Triggers option if the input and output data sources are the same type (for example, Oracle, iSeries, or SQL Server) and no calculations are performed. Selecting this option results in slower performance.

You should not use currency triggers for an environment that has a different path code from the login environment.

- Clear Output Tables

Select this option to clear the output table before the table conversion runs.

- Force Row by Row Processing

Select this option to test the table conversion or to ensure that the conversion always runs in row-by-row mode.

You can test a conversion to ensure that the mapping logic is correct. In this case, specify the number of rows to process. The number of rows can be defined in the `jde.ini` under `[TCEngine]` or when you submit the table conversion.

Select this option if you know that the values in the input table will produce duplicate keys in the output, and you want only the non-duplicate keys to be inserted.

Selecting this option results in slower processing.

- Buffer Inserts To Output Tables

Select this option to improve conversion performance if the conversion does not include any event rules to process insertion errors and you are processing row by row.

7. Click Next.

## Mapping Inputs to Outputs

Click Next on the Select Outputs form to access the Mapping form.

1. Select the event on which mapping will occur.

In most cases, you use either the Row Fetched event or Format Fetched event. Use the Format Fetched event if you are working with a user defined format.

2. Click Advanced ER to include additional event rule logic in the table conversion.

The Event Rules Design form appears for the event that you selected on the Mapping form.

3. Click Map Same to allow the system to map the inputs directly to the outputs.

If the input and outputs share some of the same data, these fields are a direct map. For PeopleSoft EnterpriseOne tables, the system maps fields by data dictionary item. For foreign tables, the system maps by column name.

4. Drag inputs to outputs to manually map the inputs to the outputs.

---

**Note.** Click Delete to erase the mapping for a selected output. Click Delete All to erase the mapping for all outputs.

---

5. If multiple output files exist, select each file from the outputs list and map the appropriate input columns to the appropriate output columns.
6. To define advanced output, double-click an output column.

The Advanced Outputs form enables you to define literals, calculations, and other mappings without using Advanced ER. You can use an advanced input to add a literal value into a field. You can also define a calculation to populate an output field, such as adding two input fields together.

7. On the Advanced Outputs form, select one of these tabs and add the appropriate input:
  - Available objects  
Select the output column, select the appropriate object, and click Apply.
  - Literal  
Select the output column, enter the appropriate value, and click Apply.
  - Defaults  
Select Use Dictionary Defaults, and click Apply.  
Select this option to use the default value in the data dictionary at runtime. If no default values exist in the data dictionary, the system displays a warning message.
  - Calculation  
Click Define Calculation and then create a calculation in Expression Manager and click Apply.
8. Click Close on the Advanced Outputs form.
9. On the Mapping form, select Issue a Write for this Event? to insert a row to the selected output after performing all column mappings for this event.  
When you select Issue a Write for this Event?, the system attaches the TC Insert Row event system function. This system function is automatically inserted at the end of the event rules. You cannot move it to another area. To specify when and where a row is inserted, attach the User Insert Row system function using Advanced ER and move it to wherever you want.
10. Click Next.

### See Also

[Chapter 10, “Setting Up Table Conversions,” Understanding the Table Conversion Process Flow, page 65](#)

[Chapter 10, “Setting Up Table Conversions,” Using Event Rules in Table Conversions, page 96](#)

## Specifying Table Conversion Logging Options

Click Next on the Mapping form to access the Logging Options form.

1. Select from these options:
  - Log All Errors.  
Indicates that every error is logged, regardless of the logging option for type of operation that failed.
  - Log Every Input Record.  
Indicates that every input record should be logged in the table conversion log file.
  - Log Outputs.  
Indicates that every output record should be logged in the table conversion log file.

- Log Deletes.  
Indicates that every record deleted is logged in the table conversion log file.
- Log Updates.  
Indicates that every record updated is logged in the table conversion log file.
- Log Copy Table Actions.  
Indicates that the tables copied are logged in the table conversion log file.
- Log Details of Copy Table Actions.  
Indicates that the detailed logging of each Copy Table action is performed.
- Run in Proof Mode.  
Indicates that the conversion is run in proof mode. In proof mode, the conversion is simulated and all actions are logged, but no data is changed.

---

**Note.** Proof mode is not an absolute proof mode. In some situations, the proof output might differ from the real output. If you insert the same record twice, for example, it may seem as though it worked in proof mode but, in reality, only one of the inserts works when you run the conversion in final mode.

---

2. Click Next after selecting logging options.

## Reviewing Table Conversion Results

Click Next on the Logging Options form to access the Finish form.

1. Select one of these options:
  - Yes, create a version of this table conversion.  
If you select yes, enter the version name.
  - No, I will create a version of this table conversion later.
2. Click Finish to complete the process.
3. On the Table Conversion form, select the Table Conversion Mappings form and review the mapping.
4. Make changes as necessary by selecting the appropriate option from the View menu. If you are satisfied with the changes, click Save.
5. To review the runtime conversion mode, select Check Conversion Mode from the Conversion menu.  
A dialog box provides you with details of the conversion mode defined.
6. From the Conversion menu, select Exit.

---

## Defining File Formats for Table Conversion Input or Output

This section provides an overview of user defined formats, lists the prerequisite, and discusses how to:

- Define delimited file formats with a single format.
- Define delimited file formats with multiple formats.
- Define fixed-width file formats with a single format.

- Define fixed-width file formats with multiple formats.

## Understanding User Defined Formats

The Table Conversion tool uses user defined formats to handle fixed-width or Comma Separated Value (CSV) files in a table or text file. These files are collectively known as flat files because they do not have relationships defined like relational database tables. Usually, flat files are text files stored on a workstation or server. They are used to import or export data from applications that have no other means of interaction. For example, you might need to share information between PeopleSoft EnterpriseOne and an external application. If the external application does not support one of the same databases that PeopleSoft EnterpriseOne supports, flat files might be the only way to transfer data between the two applications.

In a flat file, records are stored as a continuous string of information. The user defined format provides instruction on how data is presented.

This table illustrates a single database record with five columns of data: Last, First, Addr, City, and Phone:

Last	First	Addr	City	Phone
Doe	John	123 Main	Anytown	5551234

This record can be formatted as a comma-delimited string:

```
Doe, John, 123 Main, Anytown, 5551234
```

This record can also be formatted as a fix-width column format where each column is 8 characters long.

## Importing and Exporting Text Files

When you select a text file for input or output in a table conversion and do not specify a path, a default path is used. Table conversions that use the default path run on any platform. If an explicit path or iSeries library name is indicated for the file, then the file is created exactly as specified. Table conversions stored in this way might not work on other platforms, depending on the nature of the file system on each platform.

The default paths on non-iSeries platforms are:

- path code\Import\file name
- path code\Export\file name

You cannot specify a default path for the iSeries. Rather, the default is always the Import or the Export directory under the path code of the input or output environment. For example, if you are running a table conversion against the DV811 environment, the path in the file system might be \E811\DV811\import\myfile.txt.

If the conversion specifies a file name that includes anything other than the file name and extension, such as \mytextfiles\myfile.txt, the conversion attempts to open the file as specified.

## Using User Defined Formats as Input

If you use user-defined input formats, add event rules on the Format Fetched event. Without these event rules, the system ignores the format, and the data from the input table is never made available to the table conversion. If the table conversion does not require event rule logic, add comments in this event in Event Rules Design.

User defined formats work with text files and tables.

Because the procedure for importing and exporting data is database-specific, consult a database administrator for details.

## Using User Defined Formats as Output

If you use user-defined output formats, add event rules on the Format Fetched event. Without these event rules, the system ignores the format and the data from the output table is never made available to the table conversion. If the table conversion does not require event rule logic, add comments in this event in Event Rules Design.

User defined formats work with text files and tables.

Because the procedure for importing and exporting data is database-specific, you should consult a database administrator for details.

## Prerequisite

Before you begin defining delimited file formats, ensure that you have selected the input files or output files.

## Defining Delimited File Formats with a Single Format

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format and then click Define Format(s).
2. On the User Defined Format - Type form, select Delimited - Characters such as commas or tabs separate each column.
3. Select Single Format - All rows have the same format and click Next.
4. On the User Defined Format - Column Delimiter form, select the delimiter that separates the columns in the file:
  - Tab
  - Comma
  - Semicolon
  - Space
  - Other
5. Select the textual qualifier that is used to enclose a string of text:
  - None  
Indicates that no character is used to qualify text in this user-defined file.
  - Single Quotations  
Indicates that a single quotation mark is used to qualify text in this user-defined file.
  - Double Quotations  
Indicates that a double quotation mark is used to qualify text in this user-defined file.
6. If the user defined format includes column headings in the first row, select *The first row contains column headers* and click Next.
7. On the User Defined Format - Column Layout form, select a format from the list of available formats.  
You define the columns for the format so that the system can parse the information from the file.
8. For each column, click Add to define the column.
9. On the New Column Properties form, modify these fields as required and click OK.
  - Name

- Length
  - Type
10. To edit an existing column, select it, click Edit and modify the properties on the Column Properties form.
  11. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select the table or business view that you want to use as a model for the user defined format.

---

**Note.** You cannot model the columns after an existing object unless the layout of the two objects match.

---

12. Click OK and then click Next.

The system copies the format from the model that you selected and places it into the column layout grid.

13. On the User Defined Format - Finish form, click Finish.

## Defining Delimited File Formats with Multiple Formats

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format and then click Define Format(s).
2. On the User Defined Format - Type form, select Delimited - Characters such as commas or tabs separate each column.
3. Select Multiple Formats - Rows are in two (2) or more formats and click Next.
4. On the User Defined Format - Column Delimiter form, select the delimiter that separates the columns in the file:
  - Tab
  - Comma
  - Semicolon
  - Space
  - Other
5. Select the textual qualifier that is used to enclose a string of text and click Next:
  - None  
Indicates that no character is used to qualify text in this user-defined file.
  - Single Quotations  
Indicates that a single quotation mark is used to qualify text in this user-defined file.
  - Double Quotations  
Indicates that a double quotation mark is used to qualify text in this user-defined file.
6. On the User Defined Format - Multiple Format Definition form, enter the number of formats that the user defined format contains.
7. To define the character length of the Form Designator Column, enter a value in the Length field and click Next.
8. On the User Defined Format - Multiple Format Names form, name each format and define the values for each using the Name and Designator columns and then click Next.

The Designator name should describe the data in the user defined formats.

For example, suppose that a text file contains purchase order information. Lines in the table have information for a whole purchase order with the first field designated as POH; lines with a first field designated as POI contain information about individual items in the purchase order; and lines in the table with a first field designated as POT contain information about purchase order totals. In this scenario, you would enter POH as the designator of the first format, POI as the designator of the second format, and POT as the designator of the third format.

---

**Note.** You can rename the columns for each format to make it easier to remember the formats with which you are working. For example, you can rename the columns according to their function in the file, such as Header, Detail, and Total. These names appear in the Inputs list on the Mapping form of the table conversion.

---

9. On the User Defined Format - Column Layout form, select a format from the list of available formats. You define the columns for the format so that the system can parse the information from the file.

---

**Note.** To move rows, including the Format Designator, select the row and drag it to a new location.

---

10. To add additional columns, click Add to define the column.
11. On the New Column Properties form, modify these fields as required and click OK:
  - Name
  - Length
  - Type
12. To edit an existing column, select it, click Edit, and change the properties on the Column Properties form.
13. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select the table or business view that you want to use as a model for the user defined format.

---

**Note.** You cannot model the columns after an existing object unless the layout of the two objects match.

---

14. Click OK and then click Next.

The system copies the format from the model that you selected and places it into the column layout grid.

15. On the User Defined Format - Finish form, click Finish.

## Defining Fixed-Width File Formats with a Single Format

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).
2. On the User Defined Format - Type form, select Fixed Width - Columns are aligned at specific positions with specific widths.
3. Select Single Format - All rows have the same format and click Next.
4. On the User Defined Format - Column Layout form, click Add to define the column layout.
5. On the New Column Properties form, modify these fields as required, and click OK:
  - Name

- Start
  - Length
  - Type
6. To edit an existing column, select it, click Edit, and change the properties on the Column Properties form.
  7. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select the table or business view that you want to use as a model for the user defined format.

---

**Note.** You cannot model the columns after an existing object unless the layout of the two objects match.

---

8. Click OK and then click Next.  
The system copies the format from the model that you selected and places it into the column layout grid.
9. On the User Defined Format - Finish form, click Finish.

## Defining Fixed-Width File Formats with Multiple Formats

Access the Select Input form or the Select Outputs form.

1. Select User Defined Format, and then click Define Format(s).
2. On the User Defined Format - Type form, select Fixed Width - Columns are aligned at specific positions with specific widths.
3. Select Multiple Formats - Rows are in two (2) or more formats and click Next.
4. On the User Defined Format - Multiple Format Definition form, enter the number of formats that the user defined format contains and click Next.

To define the character length of the Designator column, complete the Length field.

5. On the User Defined Format - Multiple Format Names form, enter a name for each format in the Name column and then click Next.

The Designator name should describe the data in the user defined formats.

For example, suppose that a text file contains purchase order information. Lines in the table have information for a whole purchase order with the first field designated as POH; lines with a first field designated as POI contain information about individual items in the purchase order; and lines in the table with a first field designated as POT contain information about purchase order totals. In this scenario, you would enter POH as the designator of the first format, POI as the designator of the second format, and POT as the designator of the third format.

---

**Note.** You can rename the columns for each format to make it easier to remember the formats with which you are working. For example, you can rename the columns according to their function in the file, such as Header, Detail, and Total. These names appear in the Inputs list on the Mapping form of the table conversion.

---

6. On the User Defined Format - Column Layout form, select a format from the list of available formats.  
You define the columns for the format so that the system can parse the information from the file.
7. To add additional columns, click Add to define the column.
8. On the New Column Properties form, modify these fields as required and click OK:
  - Name

- Start
  - Length
  - Type
9. To edit an existing column, select it, click Edit, and change the properties on the Column Properties form.
  10. To model the columns after an existing table, business view, or foreign table, click Model, select the appropriate tab, and then select the table or business view that you want to use as a model for the user defined format.

---

**Note.** You cannot model the columns after an existing object unless the layout of the two objects match.

---

11. Click OK and then click Next.

The system copies the format from the model that you selected and places it into the column layout grid.

12. On the User Defined Format - Finish form, click Finish.

### See Also

[Chapter 10, “Setting Up Table Conversions,” Reviewing Table Conversion Results, page 76](#)

[Chapter 10, “Setting Up Table Conversions,” Using Event Rules in Table Conversions, page 96](#)

[Chapter 11, “Running Table Conversions,” Understanding Table Conversion Processing, page 99](#)

---

## Copying Data

This section provides an overview of copying data, lists the prerequisites, and discusses how to define data copy actions.

### Understanding Copying Data

You use the Data Copy option in the Table Conversion director to copy tables from one environment or data source to another. You can also import a copy table script to use in the conversion.

### Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you have:

- Reviewed the Defining External Data for Table Conversions task.
- Reviewed the Defining Input and Output Environments for Table Conversions task.
- Reviewed the Specifying Table Conversions Logging Options task.
- Reviewed the Reviewing Table Conversion Results task.

See [Chapter 10, “Setting Up Table Conversions,” Defining External Data for Table Conversions, page 70](#).

See [Chapter 10, “Setting Up Table Conversions,” Defining Input and Output Environments for Table Conversions, page 71](#).

See [Chapter 10, “Setting Up Table Conversions,” Specifying Table Conversion Logging Options, page 75](#).

See [Chapter 10, “Setting Up Table Conversions,” Reviewing Table Conversion Results, page 76.](#)

## Defining Data Copy Actions

Access the Table Conversion Director.

1. Select Data Copy and click Next.
2. On the External Data form, select a processing option template and define a data structure as appropriate and click Next.
3. On the Select Environments form, select an input environment and an output environment and click Next.
4. On the Select Actions form, enter the name of a table in the Table field of the table.

If you want to copy a single table, select *<Literal>* and enter the name of that table on the Single Value tab. If you do not know the name of the table that you want to copy, use the *<Find a Table>* option.

5. On the Select Actions form, modify these fields as necessary:

When you enter the name of a table, the system automatically populates the remaining fields for you.

- To Table

Enter either the last table in a range of tables to be copied or leave the field blank if you are copying a single table.

- Source Type

Select Data Source if the input and output sources are data sources. Select Environment if the input and output sources are environments.

If Data Source is selected, the system retrieves table definitions from the specifications in the login environment.

If Environment is selected, the system uses the input and output environments to locate data and specifications for the tables. This enables the specifications to be different in the input and output environment but the data is copied.

- Input Source

The input source is the data source or environment from which the inputs is read.

- Output Source

The output source is the data source or environment where the output is written.

- Create

If you select *<If Table Exists>*, the system creates the table in the output and runs the conversion only if both the table specifications and the actual table exist in the input.

If you select *<Yes>*, the system creates the table in the output. If the table already exists in the output, the system deletes it and creates a new table.

If you select *<No>*, the system assumes the table already exists in the output and does not create it.

- Clear

If you select *<If Table Exists>*, the system clears the table in the output only if it exists in the input.

If you select *<Yes>*, the system deletes all rows in the output table before copying the table.

If you select *<No>*, the output table will not be cleared.

---

**Note.** If you select to not clear the output table the conversion might result in key conflicts.

---

- Copy

If you select *<Yes>*, the system copies the data from the input table to the output table using Map Same.

If you select *<No>*, no data is copied.

- Owner ID

- Owner Pwd

If the data source requires an owner ID and password, enter them here. If you leave these fields blank, the system enters the ID and password of the login user, or *<None>* if the data source does not have security.

6. To import an existing copy table script from another location, click Import, locate the file that you want to import and click Open.

The system adds an action for each copy table item in the copy table script.

7. Click Advanced ER to add event rule logic to the copy table process and click Next.

You can use event rules to write a custom copy table script.

8. On the Logging Options form, select logging options and click Next.
9. On the Finish form, select a version option and click Finish.

### See Also

[Chapter 10, “Setting Up Table Conversions,” Using Event Rules in Table Conversions, page 96](#)

[Chapter 10, “Setting Up Table Conversions,” Reviewing Table Conversion Results, page 76](#)

[Chapter 11, “Running Table Conversions,” page 99](#)

---

## Copying Data with Table Input

This section provides an overview of copying data with table input, lists the prerequisites, and discusses how to define data copy with table input actions.

### Understanding Data Copy with Table Input

Data Copy with Table Input is similar to Data Copy, except that it also allows information for the process to come from an input table. The input table might provide information about which tables are copied, where they are copied, and so on. Data Copy with Table Input enables you to select data.

For example, suppose that you create a table that includes a table name, the next backup date, and the backup frequency. You might populate this table with a list of tables to be archived and information specifying how often they are archived. You can then use Data Copy with Table Input to select all rows in which the backup date is less than or equal to the current date, and calculate a new backup date.

## Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you:

- Review the Defining External Data for Table Conversions task.
- Review the Defining Input and Output Environments for Table Conversions task.
- Review the Defining Table Conversion Input task.
- Review the Defining Data Copy Actions task.
- Review the Specifying Table Conversions Logging Options task.
- Review the Reviewing Table Conversion Results task.

See [Chapter 10, “Setting Up Table Conversions,” Defining External Data for Table Conversions, page 70.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Input and Output Environments for Table Conversions, page 71.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Table Conversion Input, page 71.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Data Copy Actions, page 83.](#)

See [Chapter 10, “Setting Up Table Conversions,” Specifying Table Conversion Logging Options, page 75.](#)

See [Chapter 10, “Setting Up Table Conversions,” Reviewing Table Conversion Results, page 76.](#)

## Defining Copy with Table Input Actions

Access the Table Conversion Director.

1. Select Data Copy with Table Input and click Next.
2. On the External Data form, select a processing option template and define a data structure as appropriate and click Next.
3. On the Select Environments form, select an input environment and an output environment and click Next.
4. On the Select Input form, select a table, business view, or foreign table and click Next.
5. On the Sequencing form, define data sequencing as appropriate and click Next.
6. On the Data Selection form, define data selection as appropriate and click Next.
7. On the Table Options form, select Run Currency Triggers if appropriate and click Next.

Select this option if a PeopleSoft EnterpriseOne table included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option and the source or destination fields are currency fields used in a calculation, you might receive unexpected results.

Do not select Run Currency Triggers if the input and output data sources are the same type (for example, Oracle, iSeries, or SQL Server) and no calculations are being performed. Selecting this option results in slower processing.

You should not use currency triggers for an environment that has a different path code than the login environment.

8. On the Select Actions form, enter the name of a table in the Table field, modify the remaining fields as appropriate and click Next.

9. To import an existing copy table script from another location, click Import, locate the file that you want to import and click Open.
10. Click Advanced ER to add event rule logic to the copy table process and click Next.
11. On the Logging Options form, select logging options and click Next.
12. On the Finish form, select a version option and click Finish.

---

## Deleting Groups of Records

This section provides an overview of batch deletion, lists the prerequisites, and discusses how to define batch deletion actions.

### Understanding Batch Deletion

The Batch Delete option enables you to delete a range of records from a PeopleSoft EnterpriseOne input table or foreign table based on selection criteria that you define. For example, you can set up a batch delete table conversion that deletes records in an input table that do not contain valid data. You can also set up table conversions that delete all records from a particular table.

#### Example: Creating Purge Programs as a Batch Delete

This example illustrates how you can create a table conversion to delete records from the input environment. Designing purge programs as batch-deletes enables you to purge records with control and accuracy. You can archive the purged data or remove it from the system permanently. The archiving process is explained in this example. Before you start this example, create a handle for the table.

See [Chapter 4, “Working with Table I/O,” Understanding Handles, page 22.](#)

1. On the Introduction form of the Table Conversion director, select Batch Delete and click Next.
2. On the External Data form, select the *Purge Processing Option (T42000P)* processing option template and click Next.
3. On the Select Environment form, select *<LOGIN ENV>* as the source environment, select Force Version to Override Input Environment, and click Next.
 

The Force Version to Override Input Environment option ensures that the person who runs the purge program provides a valid source environment from which to run the batch-delete.
4. On the Select Input form, drag the *User Defined Code Types (F0004)* table to the Description column and click Next.
5. On the Table Options form, select Run Currency Triggers and click Next.
6. On the Data Selection form, define data selection as *Where IC UCD1 (User Defined Code - Class Code 1) is equal to Null* and click Next.
7. On the Logging Options form, do not select any logging options and click Next.
8. On the Finish form, select *Yes, create a version of this table conversion* and enter *VER0001* as the version name.
9. On the warning form that indicates that the table conversion needs to be saved, click OK.
10. On the Selection for Batch Delete form, select the Process Begin event and click Advanced ER.
11. In Event Rules Design, add these event rule variables with a report scope:

- *FXXXXHandle\_HFXXXX*
- *szArchiveDataSource\_DATS*
- *szPurgeDataSource\_DATS*
- *szErrorCode\_DTAI*
- *cRenameFlag\_EV01*
- *mnErrorNumber\_MATH01*

12. Enter these begin process event rules from R42119P, then save the event rules and exit Event Rules Design:

```

0001 // Check to see if the purged data is being archived
0002 If PO cArchiveRecords is equal to "1"
0003 // If the environment processing option is blank, stop processing.
0004 If PO szArchiveEnvironmentName is equal to <Blank>
    Or PO szArchiveEnvironmentName is equal to <Null>
0005 Stop Conversion Processing("The archive environment is invalid.")
0006 Else
0007 // Check to make sure that the archive environment and data source is not the
0008 // same as the input environment and data source
0009 If PO szArchiveEnvironmentName is equal to SL SourceEnvironment
0010 Stop Conversion Processing("The source and archive environments are the same")
0011 Else
0012 //Get the data source of the archive environment
0013 Get and validate the data source for an environment/table (B98700)
    PO szArchiveEnvironmentName -> szEnvironment
    "FXXXX" -> szTableName (Replace x's with the name of your table.)
    VA rpt_szArchiveDataSource_DATS <- szDataSource
    VA rpt_szErrorCode_DTAI <- szErrorDataItem
    VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0014 //SAR #4337575 - B98700 has been modified to return an error if the table is
0015 //not found in the data source. Since we do not want the table to exist in the
0016 //archive environment, we need to bypass that error condition.
0017 If VA rpt_szErrorCode_DTAI is not equal to <Blank>
    And VA rpt_szErrorCode_DTAI is not equal to <Null>
    And VA rpt_szErrorCode_DTAI is not equal to "072W"
0018 Stop Conversion Processing("No data source was found for the
archive environment")
0019 Else
0020 //Get the data source of the source environment
0021 Get and validate the data source for an environment/table (B98700)
    SL SourceEnvironment -> szEnvironment
    "FXXXX" -> szTableName (Replace x's with the name of your table.)
    VA rpt_szPurgeDataSource_DATS <- szDataSource
    VA rpt_szErrorCode_DTAI <- szErrorDataItem
    VA rpt_mnErrorNumber_MATH01 <- mnErrorNumber
0022 If VA rpt_szErrorCode_DTAI is equalto "072W"
0023 Stop Conversion Processing ("The table could not be found
in the source environment.")
0024 Else

```

```

0025     If VA rpt_szErrorCode_DTAI is not equal to <Blank>
           And VA rpt_szErrorCode_DTAI is not equal to <Null>
0026     Stop Conversion Processing ("No data source was found for
the source environment.")
0027     End If
0028     End If
0029     If VA rpt_szArchiveDataSource_DATS is equal to VA rpt_szPurgeDataSource_DATS
0030     Stop Conversion Processing ("The source and archive
environments have the same data source")
0031     Else
0032     // Open a table with the same table name in the output
environment. The table
0033     // will be renamed later if the table name processing option was populated.
0034     //Replace the X's in the following statement to reflect the
name of your table.
0035     Copy Table Environment("FXXXX", <None>, SL SourceEnvironment,
PO szArchiveEnvironmentName, <Yes>, <Yes>, <No>, <None>, <None>, <Null>,
<Null>)
0036     // Open a handle to the archive table
0037     // Replace the X's in the following statements to reflect the
name of your handle and table.
0038     VA rpt_FXXXXHandle_HFXXXX = FXXXX.Open Handle
0039     If VA rpt_FXXXXHandle_HFXXXX is equal to <Null>
0040     Stop Conversion Processing ("Failed to open FXXXX in the
archive environment")
0041     End If
0042     End If
0043     End If
0044     End If
0045 End If
0046 End If

```

---

**Note.** Map all event rule variables, even if you do not use every value.

---

In this example, the system writes log messages to the JDE.log and JDEDEBUG.log files on Stop Conversion Processing.

13. On Selection for Batch Delete, select the Row Fetched event and click Advanced ER.
14. On Event Rules Design, enter these row fetched event rules from R42119P, then save the event rules and exit Event Rules Design:

```

0001 // If we are archiving the purged records, write the record to the archive table
0002 //Replace the X's in the following statements to reflect the name of
your handle and table. Map the fields included in your table.
0003 If PO cArchiveRecords is equal to "1"
0004 FXXXX(VA rpt_FXXXXHandle_HFXXXX).Insert
           IC Order Company(Order Number) -> TK Order Company(Order Number)
           IC Document(Order No, Invoice, etc.) -> TK Document(Order No, Invoice, etc.)
           IC Order Type -> TK Order Type

```

IC Line Number -> TK Line Number  
 IC Order Suffix -> TK Order Suffix  
 IC Business Unit -> TK Business Unit  
 IC Company -> TK Company  
 IC Document Company(Original Order) -> TK Document Company(Original Order)  
 IC Original Order Number -> TK Original Order Number  
 IC Original Order Type -> TK Original Order Type  
 IC Original Line Number -> TK Original Line Number  
 IC Company-Key (Related Order) -> TK Company-Key (Related Order)  
 IC Related PO/SO/WO Number -> TK Related PO/SO/WO Number  
 IC Related PO/SO/WO Order Type -> TK Related PO/SO/WO Order Type  
 IC Related PO/SO Line Number -> TK Related PO/SO Line Number  
 IC Agreement Number-Distribution -> TK Agreement Number-Distribution  
 IC Agreement Supplement-Distribution -> TK Agreement Supplement-Distribution  
 IC Address Number -> TK Address Number  
 IC Address Number-Ship To -> TK Address Number-Ship To  
 IC Address Number-Parent -> TK Address Number-Parent  
 IC Date-Requested -> TK Date-Requested  
 IC Date-Order/Transaction -> TK Date-Order/Transaction  
 IC Date-Scheduled Pick -> TK Date-Scheduled Pick  
 IC Date-Actual Ship Date -> TK Date-Actual Ship Date  
 IC Date-Invoice -> TK Date-Invoice  
 IC Date-Cancel -> TK Date-Cancel  
 IC Date-For G/L(and Voucher) -> TK Date-For G/L(and Voucher)  
 IC Date-Promised Delivery -> TK Date-Promised Delivery  
 IC Date-Price Effective Date -> TK Date-Price Effective Date  
 IC Date-Promised Shipment -> TK Date-Promised Shipment  
 IC Reference -> TK Reference  
 IC Reference 2 -> TK Reference 2  
 IC Item Number-Short -> TK Item Number-Short  
 IC 2nd Item Number -> TK 2nd Item Number  
 IC 3rd Item Number -> TK 3rd Item Number  
 IC Location -> TK Location  
 IC Lot/Serial Number -> TK Lot/Serial Number  
 IC From Grade -> TK From Grade  
 IC Thru Grade -> TK Thru Grade  
 IC From Potency -> TK From Potency  
 IC Thru Potency -> TK Thru Potency  
 IC Days Before Expiration -> TK Days Before Expiration  
 IC Description -> TK Description  
 IC Description-Line 2 -> TK Description-Line 2  
 IC Line Type -> TK Line Type  
 IC Status Code-Next -> TK Status Code-Next  
 IC Status Code-Last -> TK Status Code-Last  
 IC Business Unit - Header -> TK Business Unit - Header  
 IC Item Number - Related (Kit) -> TK Item Number - Related (Kit)  
 IC Kit Master Line Number -> TK Kit Master Line Number  
 IC Component Line Number -> TK Component Line Number  
 IC Related Kit Component -> TK Related Kit Component  
 IC Number of Component Per Parent -> TK Number of Component Per Parent

IC Sales Catalog Section -> TK Sales Catalog Section  
 IC Sub Section -> TK Sub Section  
 IC Sales Category Code 3 -> TK Sales Category Code 3  
 IC Sales Category Code 4 -> TK Sales Category Code 4  
 IC Sales Category Code 5 -> TK Sales Category Code 5  
 IC Commodity Class -> TK Commodity Class  
 IC Commodity Sub Class -> TK Commodity Sub Class  
 IC Supplier Rebate Code -> TK Supplier Rebate Code  
 IC Master Planning Family -> TK Master Planning Family  
 IC Purchasing Category Code 5 -> TK Purchasing Category Code 5  
 IC Unit of Measure as Input -> TK Unit of Measure as Input  
 IC Units-Order/Transaction Quantity -> TK Units-Order/Transaction Quantity  
 IC Quantity Shipped -> TK Quantity Shipped  
 IC Units-Qty Backordered/Held -> TK Units-Qty Backordered/Held  
 IC Units-Quantity Canceled/Scrapped -> TK Units-Quantity Canceled/Scrapped  
 IC Units-Future Quantity Committed -> TK Units-Future Quantity Committed  
 IC Units-Open -> TK Units-Open  
 IC Units-Shipped to Date -> TK Units-Shipped to Date  
 IC Units-Relieved -> TK Units-Relieved  
 IC Committed (H/S) -> TK Committed (H/S)  
 IC Other Quantity (1/2) -> TK Other Quantity (1/2)  
 IC Amount-Price per Unit -> TK Amount-Price per Unit  
 IC Amount-Extended Price -> TK Amount-Extended Price  
 IC Amount-Open -> TK Amount-Open  
 IC Price Override Code -> TK Price Override Code  
 IC Temporary Price (Y/N) -> TK Temporary Price (Y/N)  
 IC Unit of Measure-Entered for Unit Price -> TK Unit of Measure-  
 Entered for Unit Price  
 IC Amount-List Price -> TK Amount-List Price  
 IC Amount-Unit Cost -> TK Amount-Unit Cost  
 IC Amount-Extended Cost -> TK Amount-Extended Cost  
 IC Cost Override Code -> TK Cost Override Code  
 IC Extended Cost-Transfer -> TK Extended Cost-Transfer  
 IC Print Message -> TK Print Message  
 IC Payment Terms Code -> TK Payment Terms Code  
 IC Payment Instrument -> TK Payment Instrument  
 IC Based on Date -> TK Based on Date  
 IC Discount-Trade -> TK Discount-Trade  
 IC Trade Discount (Old) -> TK Trade Discount (Old)  
 IC Price and Adjustment Schedule -> TK Price and Adjustment Schedule  
 IC Item Price Group -> TK Item Price Group  
 IC Pricing Category Level -> TK Pricing Category Level  
 IC Discount %-Cash -> TK Discount %-Cash  
 IC Document Company -> TK Document Company  
 IC Document (Voucher, Invoice, etc.) -> TK Document (Voucher, Invoice, etc.)  
 IC Document Type -> TK Document Type  
 IC Document-Original -> TK Document-Original  
 IC Document Type-Original -> TK Document Type-Original  
 IC Document Company-Original -> TK Document Company-Original  
 IC Pick Slip Number -> TK Pick Slip Number

IC Delivery Number -> TK Delivery Number  
 IC Sales Taxable(Y/N) -> TK Sales Taxable(Y/N)  
 IC Tax Rate/Area -> TK Tax Rate/Area  
 IC Tax Expl Code 1 -> TK Tax Expl Code 1  
 IC Associated Text -> TK Associated Text  
 IC Priority-Processing -> TK Priority-Processing  
 IC Printed Code -> TK Printed Code  
 IC Backorders Allowed (Y/N) -> TK Backorders Allowed (Y/N)  
 IC Substitutes Allowed (Y/N) -> TK Substitutes Allowed (Y/N)  
 IC Partial Line Shipments Allowed (Y/N) -> TK Partial Line Shipments Allowed (Y/N)  
 IC Line of Business -> TK Line of Business  
 IC End Use -> TK End Use  
 IC Duty Status -> TK Duty Status  
 IC Nature of Transaction -> TK Nature of Transaction  
 IC Primary/Last Supplier Number -> TK Primary/Last Supplier Number  
 IC Carrier Number -> TK Carrier Number  
 IC Mode of Transport -> TK Mode of Transport  
 IC Route Code -> TK Route Code  
 IC Stop Code -> TK Stop Code  
 IC Zone Number -> TK Zone Number  
 IC Container I.D. -> TK Container I.D.  
 IC Freight Handling Code -> TK Freight Handling Code  
 IC Shipping Commodity Class -> TK Shipping Commodity Class  
 IC Shipping Conditions Code -> TK Shipping Conditions Code  
 IC Serial Number-Lot -> TK Serial Number-Lot  
 IC Unit of Measure-Primary -> TK Unit of Measure-Primary  
 IC Units-Primary Quantity Ordered -> TK Units-Primary Quantity Ordered  
 IC Unit of Measure-Secondary -> TK Unit of Measure-Secondary  
 IC Units-Secondary Quantity Ordered -> TK Units-Secondary Quantity Ordered  
 IC Unit of Measure-Pricing -> TK Unit of Measure-Pricing  
 IC Unit Weight -> TK Unit Weight  
 IC Weight Unit of Measure -> TK Weight Unit of Measure  
 IC Unit Volume -> TK Unit Volume  
 IC Volume Unit of Measure -> TK Volume Unit of Measure  
 IC Reprice (Basket Price) Category -> TK Reprice (Basket Price) Category  
 IC Order Reprice Category -> TK Order Reprice Category  
 IC Order Repriced Indicator -> TK Order Repriced Indicator  
 IC Costing Method-Inventory -> TK Costing Method-Inventory  
 IC G/L Offset -> TK G/L Offset  
 IC Century -> TK Century  
 IC Fiscal Year -> TK Fiscal Year  
 IC Inter Branch Sales -> TK Inter Branch Sales  
 IC On Hand Updated -> TK On Hand Updated  
 IC Configurator Print Flag -> TK Configurator Print Flag  
 IC Sales Order Status 04 -> TK Sales Order Status 04  
 IC Substitute Item Indicator -> TK Substitute Item Indicator  
 IC Preference Commitment Indicator -> TK Preference Commitment Indicator  
 IC Ship date (PDDJ) overridden -> TK Ship date (PDDJ) overridden  
 IC Price Adjustment Line Indicator -> TK Price Adjustment Line Indicator  
 IC Price Adj. History Indicator -> TK Price Adj. History Indicator

IC Preference Production Allocation -> TK Preference Production Allocation  
 IC Transfer/Direct Ship/Intercompany Flag -> TK Transfer/Direct Ship/  
 Intercompany Flag  
 IC Deferred entries flag -> TK Deferred entries flag  
 IC Euro Conversion Status Flag -> TK Euro Conversion Status Flag  
 IC Sales Order Status 14 -> TK Sales Order Status 14  
 IC Sales Order Status 15 -> TK Sales Order Status 15  
 IC Apply Commission(Y/N) -> TK Apply Commission(Y/N)  
 IC Commission Category -> TK Commission Category  
 IC Reason Code -> TK Reason Code  
 IC Gross Weight -> TK Gross Weight  
 IC Gross Weight Unit of Measure -> TK Gross Weight Unit of Measure  
 IC Subledger-G/L -> TK Subledger-G/L  
 IC Subledger Type -> TK Subledger Type  
 IC Code-Location Tax Status -> TK Code-Location Tax Status  
 IC Price Code 1 -> TK Price Code 1  
 IC Price Code 2 -> TK Price Code 2  
 IC Price Code 3 -> TK Price Code 3  
 IC Status-In Warehouse -> TK Status-In Warehouse  
 IC Work Order Freeze Code -> TK Work Order Freeze Code  
 IC Send Method -> TK Send Method  
 IC Currency Code-From -> TK Currency Code-From  
 IC Currency Conversion Rate-Spot Rate -> TK Currency Conversion Rate-Spot Rate  
 IC Amount-List Price per Unit -> TK Amount-List Price per Unit  
 IC Amount-Foreign Price per Unit -> TK Amount-Foreign Price per Unit  
 IC Amount-Foreign Extended Price -> TK Amount-Foreign Extended Price  
 IC Amount-Foreign Unit Cost -> TK Amount-Foreign Unit Cost  
 IC Amount-Foreign Extended Cost -> TK Amount-Foreign Extended Cost  
 IC User Reserved Code -> TK User Reserved Code  
 IC User Reserved Date -> TK User Reserved Date  
 IC User Reserved Amount -> TK User Reserved Amount  
 IC User Reserved Number -> TK User Reserved Number  
 IC User Reserved Reference -> TK User Reserved Reference  
 IC Transaction Originator -> TK Transaction Originator  
 IC User ID -> TK User ID  
 IC Program ID -> TK Program ID  
 IC Work Station ID -> TK Work Station ID  
 IC Date-Updated -> TK Date-Updated  
 IC Time of Day -> TK Time of Day  
 IC Manufacturing Variance Accounting Flag -> TK Manufacturing  
 Variance Accounting Flag  
 IC Sales Order Status 17 -> TK Sales Order Status 17  
 IC Sales Order Status 18 -> TK Sales Order Status 18  
 IC Sales Order Status 19 -> TK Sales Order Status 19  
 IC Sales Order Status 20 -> TK Sales Order Status 20  
 IC Integration Reference 01 -> TK Integration Reference 01  
 IC Integration Reference 02 -> TK Integration Reference 02  
 IC Integration Reference 03 -> TK Integration Reference 03  
 IC Integration Reference 04 -> TK Integration Reference 04  
 IC Integration Reference 05 -> TK Integration Reference 05

```

IC Source of Order -> TK Source of Order
IC Reference -> TK Reference
IC Demand Unique Key ID -> TK Demand Unique Key ID
IC Pull Signal -> TK Pull Signal
IC Release Number -> TK Release Number
IC Scheduled Shipment Time -> TK Scheduled Shipment Time
IC Time-Release -> TK Time-Release
IC Date-Release -> TK Date-Release
IC Requested Delivery Time -> TK Requested Delivery Time
IC Actual Shipment Time -> TK Actual Shipment Time
IC Time-Original Promised Delivery -> TK Time-Original Promised Delivery
IC Time-Scheduled Pick -> TK Time-Scheduled Pick
IC Time-Future Time 2 -> TK Time-Future Time 2
IC Cross-Docking Flag -> TK Cross-Docking Flag
IC Cross-Docking Priority for Sales Orders -> TK Cross-Docking
Priority for Sales Orders
IC Dual Unit of Measure Item -> TK Dual Unit of Measure Item
IC Buying Segment Code -> TK Buying Segment Code
IC Current Buying Segment Code -> TK Current Buying Segment Code
IC Change Order Number -> TK Change Order Number
IC Address Number-Deliver To -> TK Address Number-Deliver To
IC Pending Approval Flag -> TK Pending Approval Flag
IC Revision Reason -> TK Revision Reason
IC Matrix Control Line Number -> TK Matrix Control Line Number
IC Shipment Number -> TK Shipment Number
IC Promised Delivery Time -> TK Promised Delivery Time
IC Project Number -> TK Project Number
IC Sequence Number -> TK Sequence Number
IC Item Revision Level -> TK Item Revision Level
IC Hold Orders Code -> TK Hold Orders Code
IC Business Unit-Header -> TK Business Unit-Header
IC Business Unit-Demand -> TK Business Unit-Demand
IC Currency Code-Base -> TK Currency Code-Base
IC Document Line Number-Original -> TK Document Line Number-Original
IC Date-Original Promised Delivery -> TK Date-Original Promised Delivery
IC Cross Dock Order Company(OrderNumber) -> TK Cross Dock Order
Company(OrderNumber)
IC Cross Dock Order No -> TK Cross Dock Order No
IC Cross Dock Order Type -> TK Cross Dock Order Type
IC Cross Dock Line Number -> TK Cross Dock Line Number
IC Cross Dock Order Suffix -> TK Cross Dock Order Suffix
IC Port of Entry or Exit -> TK Port of Entry or Exit
IC Payment Terms Override Code -> TK Payment Terms Override Code
IC Buyer Number -> TK Buyer Number
IC Promotion ID -> TK Promotion ID
IC Asset Item Number -> TK Asset Item Number
IC Parent Number -> TK Parent Number
0005 // Do not delete the record if the insert to the archive table failed.
0006 If SV Error_Status is not equal to CO ERROR
0007 Delete Current Input Row

```

```

0008 End If
0009 Else
0010 Delete Current Input Row
0011 End If

```

---

**Note.** Ensure that you have mapped all parameters to a field, even if you do not use every value.

---

15. On Selection for Batch Delete, select the Process End event and click Advanced ER.
16. On Event Rules Design, enter these process end event rules from R42119P, then save the event rules and exit Event Rules Design:

Ensure that you have mapped all parameters to a field, even if you do not use every value.

```

0001 If PO cArchiveRecords is equal to "1"
0002 // Close the table
0003 //Replace the X's in the following statement to reflect the
name of your handle and table.
0004 FXXXX(VA rpt_FXXXXHandle_HFXXXX).Close
0005 // If the data was archived and the table name processing option was
populated,
0006 // rename the table.
0007 If PO szArchiveTableName is not equal to <Blank>
And PO szArchiveTableName is not equal to <Null>
0008 Rename Table (B000202)
"XXXX" -> szOldTableName
PO szArchiveTableName -> szNewTableName
"<Blank>" -> szTableOwnerID
"<Blank>" -> szPassword
VA rpt_szArchiveDataSource_DATS -> szDataSource
VA rpt_cRenameFlag_EV01 <- cRenameTableSuccessful
0009 End If
0010 End If

```

---

**Note.** Ensure that you have mapped all parameters to a field, even if you do not use every value.

---

## See Also

[Chapter 10, "Setting Up Table Conversions," Deleting Groups of Records, page 86](#)

[Chapter 4, "Working with Table I/O," Understanding Handles, page 22](#)

## Prerequisites

Before you begin defining the input and output environments for copying data, ensure that you:

- Review the Defining External Data for Table Conversions task.
- Review the Defining Input and Output Environments for Table Conversions task.
- Review the Defining Table Conversion Input task.
- Review the Defining Data Copy Actions task.

- Review the Specifying Table Conversions Logging Options task.
- Review the Reviewing Table Conversion Results task.

See [Chapter 10, “Setting Up Table Conversions,” Defining External Data for Table Conversions, page 70.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Input and Output Environments for Table Conversions, page 71.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Table Conversion Input, page 71.](#)

See [Chapter 10, “Setting Up Table Conversions,” Defining Data Copy Actions, page 83.](#)

See [Chapter 10, “Setting Up Table Conversions,” Specifying Table Conversion Logging Options, page 75.](#)

See [Chapter 10, “Setting Up Table Conversions,” Reviewing Table Conversion Results, page 76.](#)

## Defining Batch Deletion Actions

Access the Table Conversion Director.

1. Select Batch Delete and click Next.
2. On the External Data form, select a processing option template and define a data structure as appropriate and click Next.
3. On the Select Environments form, select a source environment and click Next.
4. On the Select Input form, drag the appropriate table to the Description column and click Next.

To delete an input table, select it and press DELETE.

---

**Note.** If you change the table, the system warns you that deleting tables removes all mappings from the table conversion.

---

5. On the Table Options form, select Run Currency Triggers if appropriate and click Next.  
 Select this option if a PeopleSoft EnterpriseOne table included in the table conversion contains currency triggers. If a table contains currency fields and you do not select this option, the system cannot determine where the decimal should be placed within a field. If you do not select the currency trigger option and the source or destination fields are currency fields used in a calculation, you might receive unexpected results.  
 Do not select Run Currency Triggers if the input and output data sources are the same type (for example, Oracle, iSeries, or SQL Server) and no calculations are being performed. Selecting this option results in slower processing.  
 You should not use currency triggers for an environment that has a different path code than the login environment.
6. On the Data Selection form, define data selection as appropriate.
7. Select the *Row Fetched* event when deleting rows; otherwise, no records are deleted.  
 When you run the table conversion, the system fetches the rows one at a time, runs the conversion for each row, and deletes the record from the input.
8. Click Advanced ER to insert additional event rule logic.
9. Verify that the Delete All Selected Records option is selected and click Next.  
 This option inserts the Delete Current Input Row system function into event rules.
10. On the Logging Options form, select logging options and click Next.

11. On the Finish form, select a version option and click Finish.

---

## Using Event Rules in Table Conversions

This section provides an overview of event rules in table conversions and discusses how to create event rules in table conversions.

### Understanding Event Rules in Table Conversions

You can use event rules to build complex functional logic into table conversions. For example, you can use event rules to insert information into a table or delete rows in a table based on specific conditions.

You attach event rules to an event, such as Process Begin, Row Fetched, Format Fetched, and Process End.

Event rules in table conversions include system functions that are specific to the Table Conversion tool.

### Table Conversion System Functions

This table describes each system function that you can use in table conversion event rules:

System Function	Description
CopyTableDataSource	Use this system function to copy a table or range of tables from one data source to another. The system copies tables based on specifications in the login environment.
CopyTableEnvironment	Use this system function to copy a table or range of tables from one environment to another. The system copies tables based on specifications in the input and output environments. If the specifications differ, the system performs a map-and-drop. This means that it creates a mapping between like fields in the source and destination tables, and it ignores all other fields.
TCInsertRow	The table conversion system inserts this system function when you select the Issue a write for this event? option and it cannot be moved. This function instructs the system that data should be written to the output table.
UserInsertRow	Use this system function to specify when and where a row is inserted into the specified output table.
DeleteCurrentInputRow	Use this system function to delete the current record from the input table.
UpdateCurrentInputRow	Use this system function to update the current record in the input table after it has been modified.

System Function	Description
SetSelectionAppendFlag	Use this system function to determine whether selection criteria added by the system function SetUserSelection should append to or replace the existing selection criteria on the input table.
SetUserSelection	Use this system function to conditionally modify data selection on the input table. Call SetSelectionAppendFlag before calling SetUserSelection to indicate whether to replace or append to the existing data selection.

## See Also

Chapter 10, “Setting Up Table Conversions,” *Understanding the Table Conversion Process Flow*, page 65

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Event Rules and System Functions*, “Understanding Events, Event Rules, and Runtime Processing”

## Creating Event Rules in Table Conversions

Access a table conversion form that includes an Advanced ER button.

1. From the Events drop-down list, select the event to which you want to attach event rules and click Advanced ER.
2. On the Event Rules Design form, click any of these buttons to define specific business logic:
  - **Assignment**  
Enables you to assign a fixed value, a field, or a mathematical expression to a field or variable.
  - **If\While**  
Enables you to create If and While logic statements, which are conditional instructions for event rule logic.
  - **Business Function**  
Enables you to attach an existing PeopleSoft EnterpriseOne business function to an event. Business functions can retrieve a next number value for a new customer or convert Julian calendar dates to month, day, and year.
  - **System Function**  
Enables you to attach an existing PeopleSoft EnterpriseOne system function, such as Copy Table Environment or User Insert Row.
  - **Variables**  
Enables you to create event rule variables, using existing data items, to store values for additional processing. Event rule variables can be used to accumulate totals, record the number of records that are read, and so on.
  - **Else**  
Enables you to create Else logic statements. An Else statement is automatically inserted after an If statement.

- Table I/O

Enables you to open tables in the input, output, or login environment. Table I/O enables you to read data from tables other than the input table and use the data to create an output record. For example, suppose that you create a table conversion that loops through records in the F0101 table. The table conversion copies the records to another table, loops back through the records to find each customer that has a specific employee as a contact, and copies the information to the output table.

- Report Interconnect

Enables you to connect a batch process or report to the table conversion.

3. Define the logic, save the event rules and exit Event Rules Design.

### **See Also**

Chapter 4, “Working with Table I/O,” Working With Table I/O, page 23

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Event Rules and System Functions, “Using Event Rules Design”*

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: APIs and Business Functions, “Using Business Functions”*

# CHAPTER 11

## Running Table Conversions

This chapter provides an overview of table conversion processing and discusses how to:

- Submit table conversions.
- Test table conversions.

---

### Understanding Table Conversion Processing

When you run a table conversion, you submit it using a batch version. To track the table conversion process, you can use the tracing feature. The tracing feature writes the details of the table conversion process to a log. You can define the trace level to control the level of detail that is included in the log. When you test a table conversion, you can select to process the conversion one row at a time, which enables you to isolate problems or unexpected results.

---

### Submitting Table Conversions

This section provides overviews of submitting table conversions and overriding table conversion properties and discusses how to:

- Override table conversion properties at runtime.
- Override table conversion properties of new versions.
- Override table conversion locations.
- Run table conversions on the web client.

### Understanding Submitting Table Conversions

When you submit a batch version to process a table conversion, you can override the table conversion properties, such as: input and output environment or trace level, and override the location where the table conversion processes.

Overriding the location where you process the table conversion enables you to process the table conversion even if the server that you normally use is inoperable.

## See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Batch Versions*, “Submitting Batch Versions,” Submitting Table Conversions From the Microsoft Windows Client

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Batch Versions*, “Modifying Properties of Batch Versions,” Accessing Properties for Versions of Table Conversions

## Understanding Overriding Table Conversion Properties

When you submit a batch version to process a table conversion, you can override the conversion properties:

- Environments.  
Select this override if you need to change the location of the input or output data.
- Data selection.  
Select this override if you need to limit the number of input records.
- Table options.  
Select this override if you need to controls how Table Conversion processes records during the conversion.
- Logging options.
- Debug logging.  
Select this override if you need to overrides the setting for debug logging as defined in the jde.ini.

### Table Conversions on the Web Client

When submitting versions of table conversions from the web client you cannot modify properties. You can select and modify processing options. You can also select these options:

- Override Location
- Override Job Queue
- Submit Version Specifications Only
- Data Selection

See *PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Batch Versions*, “Working with Batch Version Specifications,” Copying Batch Version Specifications to an Enterprise Server.

## Forms Used to Submit Table Conversions

Form Name	FormID	Navigation	Usage
Work With Batch Versions-Available Versions	W98305A	EnterpriseOne Life Cycle Tools, Report Management (GH9111), Batch Versions	Select the table conversion template.
Table Conversion Prompting	W98305D	Highlight a version on the Work With Batch Versions form and click Select.	Work with versions.
Properties	NA	Select the Properties option on the Table Conversion Prompting form and click Submit.	Review and override the environments, data selection, table options, and logging options specified within the conversion.
PeopleSoft Data Sources	W98305B	Select the Override Location option on the Table Conversion Prompting form and click Submit.	Select a data source to use as an override location.

## Overriding Table Conversion Properties at Runtime

Access the Work With Batch Versions-Available Versions form.

---

**Note.** If you override table conversion properties at runtime, the overrides are for that single process only, they are not saved with the version.

---

1. Enter the name of the table conversion in the Batch Application field and click Find.  
You can also use the visual assist in the Batch Application field to search for the correct table conversion.
2. Click a batch version and click Select.
3. On the Table Conversion Prompting form, select Properties and click Submit.
4. On the Properties form, review and override the environments, data selection, table options, and logging options specified in the table conversion.

These forms are similar to the forms in Table Conversion Design.

5. Select the Debug Logging tab to perform these steps:
  - Select Use ini settings for trace level and number of rows to process to use the jde.ini settings for the trace level and row-by-row conversion process.  
This option ensures that the system uses the settings defined in the jde.ini instead of the values entered in the Trace Level and Number of Rows fields.
  - Clear the Use ini settings for trace level and number of rows to process option to override the trace level in the jde.ini. Enter a value from 0 to 10 in the Trace Level field.
  - Clear the Use ini settings for trace level and number of rows to process option to convert a specific number of records (for example, if you want to test the table conversion). Enter the number of rows to process in the Number of Rows field.

If you enter 0 in the Number of Rows field, the system processes all rows.

This option corresponds to the StopAfterRow setting in the jde.ini file. If you enter a value here, you override any specifications in the jde.ini.

- Click OK to save the changes to the version.

The system submits the table conversion.

### See Also

[Chapter 11, “Running Table Conversions,” Testing Table Conversions, page 103](#)

## Overriding Table Conversion Properties of New Versions

Access the Work With Batch Versions-Available Versions form.

---

**Note.** If you override table conversion properties from the Tools tab, the new properties are saved with the version.

---

- Enter the name of the table conversion in the Batch Application field and click Find.  
You can also use the visual assist in the Batch Application field to search for the correct table conversion.
- Add a new version of the table conversion, complete the version information, and click OK.
- On the Table Conversion Version Design form, select the Tools tab and click Properties.
- On the Properties form, review and override the environments, data selection, table options, and logging options specified in the table conversion.

These forms are similar to the forms in Table Conversion Design.

- To use debug logging, select the Debug Logging tab to perform these steps:
  - Select Use ini settings for trace level and number of rows to process to use the jde.ini settings for the trace level and row-by-row conversion process.  
This option ensures that the system uses the settings defined in the jde.ini instead of the values entered in the Trace Level and Number of Rows fields.
  - Clear the Use ini settings for trace level and number of rows to process option to override the trace level in the jde.ini. Enter a value from 0 to 10 in the Trace Level field.
  - Clear the Use ini settings for trace level and number of rows to process option to convert a specific number of records (for example, if you want to test the table conversion). Enter the number of rows to process in the Number of Rows field.

If you enter 0 in the Number of Rows field, the system processes all rows.

This option corresponds to the StopAfterRow setting in the jde.ini file. If you enter a value here, you override any specifications in the jde.ini file.

## Overriding Table Conversion Locations

Access the Work With Batch Versions-Available Versions form.

- Enter the name of the table conversion in the Batch Application field and click Find.  
You can also use the visual assist in the Batch Application field to search for the correct table conversion.
- Click a batch version and click Select.
- On the Table Conversion Prompting form, select Override Location and click Submit.
- On the PeopleSoft Data Sources form, select the data source to use and click Select.

## Running Table Conversions on the Web Client

From EnterpriseOne Menu select Submit Job to access the Submit Job-Work With Batch Versions-Available Versions form.

1. Enter the name of the table conversion in the Batch Application field and click Find.  
The Batch Application field now reads Table Conversion.
2. Select a batch version and click Select.
3. On the Table Conversion Prompting form, select from these override options and click Submit:
  - Override Location
  - Override Job Queue
  - Submit Version Specifications Only
  - Data Selection
4. Modify the appropriate property and submit the batch version.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Development Tools: Batch Versions*, “Submitting Batch Versions,” Submitting Table Conversions From the Web Client

---

## Testing Table Conversions

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

- Set the trace level for debug logging on the workstation.
- Set the trace level for debug logging on the server.
- Force row-by-row conversion.

## Understanding Table Conversion Testing

You can test table conversions in proof mode to ensure that they run as expected without errors. You can log debug information about the conversion while it runs. You can also force the conversion to process one row at a time, which is useful if the conversion normally runs as an insert-from-select.

To log debug information about table conversions, enable tracing and set a trace level in the `jde.ini`. Use a value in the range of 0 to 10 for the trace level indicating the level of detail that you want to include in the log. If necessary, you can override the `jde.ini` settings on the batch version of a table conversion.

If you set the trace level for logging at 1, the system logs basic information about the table conversion, such as name, inputs, outputs, event rule logic, and how many rows were inserted.

If you set the trace level at 10, the system logs all information about every column in every format, including user defined formats, processing options associated with the table conversion, and all other information involved in the table conversion process. The higher you set the trace level, the more information the system supplies about the table conversion process.

When you test a table conversion, you can force row-by-row processing. You can force row-by-row processing on the workstation or on the server, when you want to test the table conversion. You can also set a trace level on the workstation or on the server, depending on where you process the table conversion. You can specify the number of rows to process in combination with forcing a row-by-row conversion.

### Difference Between Logging Options and Debug Logging

Logging options, which you define when you set up a table conversion, can log all errors that occur during the table conversion or can log all records that are copied, deleted, or updated. They can also log the details of copy table actions.

Debug logging logs more detailed information about the table conversion. This information enables you to pinpoint the exact area in the table conversion where errors occurred.

---

**Note.** If you modify the debug logging in the batch version, you override the settings in the jde.ini.

---

### Trace Levels

You can define a trace level for debug logging either on a workstation or on a server, depending on where you run the table conversion.

This table describes each trace level:

Trace Level	Logging Information
Level 1	Logs general information about the conversion, such as name, inputs, outputs, event rule logic, and how many rows were inserted.
Level 2	Logs function call traces, such as starting conversion, ending conversion, and inserting rows. Also includes all level 1 information.
Level 3	Logs the points at which event rules are executed. Also includes all level 1 and level 2 information.
Level 4	Not applicable.
Level 5	Logs the points at which jdeCallObject is executed, such as calls to business functions from event rules. Also includes all level 1, level 2, and level 3 information.
Levels 6-8	Not applicable.
Level 9	Logs the content of columns during input, event rules, and before output. Also includes all level 1, level 2, level 3, and level 5 information.
Level 10	Logs all information for all applicable levels.

---

**Important!** Do not set the trace level at 10 when running a table conversion on tables that contain large amounts of data. The system writes the data to the server, and large amounts of data can cause the server to run out of disk space. You can, however, specify the number of rows to process on the batch version properties form.

---

## Setting the Trace Level for Debug Logging on the Workstation

To define the trace level for debug logging on the workstation.

1. Open the jde.ini on the workstation.
2. To enable tracing and to set the trace level on a workstation, *add* this new information to the jde.ini:

```
[TCEngine]
_continue-
TraceLevel=n
```

where *n* is a number from 0 through 10.

3. To enable tracing and to set the trace level on a workstation, *modify* this information in the jde.ini:

```
[Debug]
Output=File
```

```
[UBE]
UBESaveLogFile=1
```

## Setting the Trace Level for Debug Logging on the Server

To define the trace level for debug logging on the server.

1. Open the jde.ini on the server.
2. To enable tracing and to set the trace level on a server, *add* this new information to the jde.ini:

```
[TCEngine]
_continue

TraceLevel=n
```

where *n* is a number from 0 to 10.

3. To enable tracing and to set the trace level on a server, *modify* this information in the jde.ini:

```
[Debug]
Output=File
```

```
KeepLogs=1
```

## Forcing Row-By-Row Conversion

To force row-by-row conversion and specify the number of rows to process:

1. Open the jde.ini file.
2. Add this new information to the jde.ini to force row-by-row conversion:

```
[TCEngine]
ForceRowByRow=1
_continue -
```

3. Beneath the [TCEngine] header, add this new information to specify the number of rows to process:

`StopAfterRow=n`

where *n* is the number of rows that you want to process.

## CHAPTER 12

# Preparing Foreign Tables for Table Conversion

This chapter provides an overview of foreign tables and discusses how to:

- Add PeopleSoft EnterpriseOne data sources.
- Add PeopleSoft EnterpriseOne environments.
- Set up default OCM mappings.

---

## Understanding Foreign Tables

Foreign tables are text files, or any other files or tables, that are not recognized by PeopleSoft EnterpriseOne software. However, they must reside in a database that is supported by PeopleSoft EnterpriseOne, which includes Oracle, Access, iSeries, and SQL Server.

Before you can work with foreign tables in Table Conversion Design, you must define the database to PeopleSoft EnterpriseOne. This definition is created when you set up an ODBC data source, or an Oracle Call Interface (OCI) data source, that points to the database where the foreign tables reside.

You must also add a data source in PeopleSoft EnterpriseOne that points to the ODBC or OCI data source that you defined. For each ODBC data source, database instance, or library that contains foreign tables, you must set up an environment. The environment points to the PeopleSoft EnterpriseOne data source, which in turn points to the database or library. In addition, you must set up an ODBC mapping from the data source to the environment.

---

**Note.** When you work with foreign tables, you must discuss database permissions with a database administrator. Your PeopleSoft EnterpriseOne user ID (or, if you are using the PeopleSoft EnterpriseOne security server feature, the database user to which it maps) might need to be changed to give you permissions to use the tables in the foreign database. Without these permissions, you cannot view the foreign tables in Table Conversion Design. Under certain conditions, the table conversion engine must create temporary tables in the output environment and requires create-and-drop permissions for the database.

---

---

## Adding PeopleSoft EnterpriseOne Data Sources

This section discusses how to add a PeopleSoft EnterpriseOne data source.

## Forms Used to Add PeopleSoft EnterpriseOne Data Sources

Form Name	FormID	Navigation	Usage
Machine Search & Select	W986115E	EnterpriseOne Life Cycle Tools, System Administration Tools, System Administration Tools (GH9011), Data Source Management, Database Data Sources	Select where the data source resides.
Work With Data Sources	W986115A	Select the appropriate machine on which the data source resides, and then click Select on the Machine Search & Select form.	Add a data source or select a data source to modify.
Data Source Revisions	W986115O	Click Add on the Work With Data Sources form.	Enter the data source name, use, type, data class, platform, server name, database name and owner.

## Adding PeopleSoft EnterpriseOne Data Sources

Access the Data Source Revisions form.

The screenshot shows the 'Data Source Revisions' form with the following fields and values:

- Data Source Name: Foreign Table - TEST
- Data Source Use: DB
- Data Source Type: O
- Data Class: L
- Platform: LOCAL
- Database Server Name: DEN-EDU003
- Object Owner ID: admin
- Database Name (TNS): Legacy

Data Source Revisions form

<b>Data Source Name</b>	A unique name that identifies the data source.
<b>Data Source Use</b>	Indicates how the data source is configured, Servers (SVR) to run UBEs and business functions or a Database (DB) to access table data.  Enter <i>DB</i> in this field to identify the data source as a database data source. Only database data sources are used when accessing data in tables.
<b>Data Source Type</b>	Indicates the type of database.
<b>Data Class</b>	A user defined code value that describes the type of tables in the database, such as business data, control tables, and data dictionary.
<b>Platform</b>	The type of physical hardware that the database resides on, such as <i>AS/400</i> , <i>Linux</i> , and <i>Sun Microsystems</i> .
<b>Database Server Name</b>	The name of the machine where the database or file resides.
<b>Object Owner ID</b>	The database table prefix or owner. This field is available for these data source types: <ul style="list-style-type: none"> <li>• SQL Server OLEDB</li> <li>• SQL Server ODBC</li> <li>• MSDE/ODBC</li> <li>• MSDE/OLEDB</li> <li>• Oracle</li> </ul>
<b>Database Name</b>	The name assigned to the database during installation, such as <i>HPDEVORAP</i> or <i>HP9000</i> .  Depending on the data source type, this field is used differently. If an Oracle data source is added, then this field contains the Oracle connect string. If a Microsoft Access, SQL*Server or Client Access data source is added, then this field contains the Windows ODBC data source name. To minimize the number of connections to SQL*Server, it is recommended that the ODBC data source name is the machine name where the SQL*Server database resides and that the catalog name is defined using the individual database name. For example, if you have two databases, DatabaseA and DatabaseB, on a machine called INTELNT, this field should contain INTELNT and the catalog name would be set to DatabaseA for one data source and DatabaseB for the other data source.  The data source name can be different from the ODBC or Oracle database name, if necessary.
<b>Database Instance</b>	Indicates the name of the server instance. This name is required for these data source types: <ul style="list-style-type: none"> <li>• SQL Server OLEDB</li> <li>• SQL Server ODBC</li> <li>• MSDE/ODBC</li> <li>• MSDE/OLEDB</li> </ul>
<b>ODBC Data Source Name</b>	Indicates the name assigned to the database during installation, such as <i>HPDEVORAP</i> or <i>HP9000</i> . This field is available for these data source types: <ul style="list-style-type: none"> <li>• Access</li> </ul>

- SQL Server ODBC
- DB2 UDB on OS/390
- DB2 UDB on OS/400
- MSDE/ODBC
- DB2 UDB on UNIX or Windows

**Schema** The database table prefix or owner. This field is available for these data source types:

- DB2 UDB on OS/390
- DB2 UDB on UNIX or Windows

**Database Alias Name** The name of the database. This field is available for these data source types:

- DB2 UDB on OS/390
- DB2 UDB on UNIX or Windows

**Library Name** The name of the database. This field is available for the DB2 UDB on OS/400 data source type.

## Adding PeopleSoft EnterpriseOne Environments

The easiest way to add an environment to PeopleSoft EnterpriseOne is to copy an existing environment. This section discusses how to add PeopleSoft EnterpriseOne Environments.

### Forms Used to Add PeopleSoft EnterpriseOne Environments

Form Name	FormID	Navigation	Usage
Work With Environments	W0094E	EnterpriseOne Life Cycle Tools, System Administration Tools, System Administration Tools, Environment Management (GH9053), Environment Master	Select an environment to copy.
Copy Environment	W0094B	Click an environment to copy and select Copy Environment from the Row menu on the Work With Environments form.	Enter a name for the new environment.

### Adding PeopleSoft EnterpriseOne Environments

Access the Work With Environments form.

1. Click the environment that most closely matches the environment that you want to create (such as the environment that you are logged in to or any other environment that you can access from your workstation), and select Copy Environment from the Row menu.

2. On the Copy Environment form, enter a name for the new environment in the New Environment field.
3. To copy only the \*PUBLIC OCM mappings of an environment, select Copy \*PUBLIC Records Only and click OK.

The Copy \*PUBLIC Records Only option is selected by default. Clear the option to copy all mappings for the environment including individual users, and \*PUBLIC.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, "Setting Up Environments," Working with Environments*

---

## Setting Up Default OCM Mappings

This section provides an overview of OCM mappings and discusses how to set up default OCM mappings.

### Understanding OCM Mapping

Map PeopleSoft EnterpriseOne objects, such as tables, by environment. When you define a default OCM mapping, select an existing environment and map the objects for that environment to the data sources where those objects exist.

You must create a default map for the TBLE object type. Create this mapping with a literal value of DEFAULT as the object name and TBLE as the object type. You must also indicate the data source where this table object resides. When you create a default map for the TBLE object type, all table objects point to the default data source unless the table has its own specific mapping. In addition, the Table Conversion tool uses this mapping for foreign tables.

Each environment must have a default map for table objects for the \*PUBLIC user profile because no inherent default location exists for table objects. If table objects do not have a default map and are not explicitly mapped by name, PeopleSoft EnterpriseOne produces a Select/Failed error message when it tries to access the tables. Additionally, the tables do not appear in the input or output forms in Table Conversion Design.

## Forms Used to Set Up Default OCM Mappings

Form Name	FormID	Navigation	Usage
Machine Search and Select	W986110D	EnterpriseOne Life Cycle Tools, System Administration Tools, System Administration Tools (GH9011), Data Source Management, Object Configuration Manager	Select a data source.
Work With Object Mappings	W986110B	Select the appropriate machine on which the data source resides, and then click Select on the Machine Search & Select form.	Add, modify, and copy mappings.
Object Mapping Revisions	W986110C	Click Add on the Work With Object Mappings form.	Enter the environment name, object name, primary data source, system role, object type, and data source mode for a new mapping and indicate whether to allow QBE for the mapping.

## Setting Up Default Mappings

Access the Machine Search and Select form.

The screenshot shows a software window titled "Object Configuration Manager - [Object Mapping Revisions]". The window has a menu bar with "File", "Edit", "Preferences", "Window", and "Help". Below the menu bar is a toolbar with icons for "OK", "Cancel", "Discard", "Apply", "Links", "Display", "OLE", and "Internet". The main area of the form contains the following fields and values:

- Environment Name: FOREIGNDE
- Object Name: DEFAULT
- Primary Data Source: Business Data Local
- System Role: \*PUBLIC
- Object Type: TBLE (with "Table Definition" text to the right)
- Data Source Mode: P (with "Primary" text to the right)
- Secondary Data Source: (empty field)
- Allow QBE: 1 (with "All QBE Allowed" text to the right)

Object Mapping Revisions form

<b>Environment Name</b>	The name of the environment where the tables reside.
<b>Object Name</b>	The name of the specific object that you are mapping. The value DEFAULT creates a default map for all objects of a specific type.
<b>Primary Data Source</b>	The name of the data source.
<b>System Role</b>	The PeopleSoft EnterpriseOne role with permissions to use these mappings. Select an individual user, group, or * PUBLIC. * PUBLIC gives all users permissions.
<b>Object Type</b>	The type of object for which you are creating a mapping.
<b>Data Source Mode</b>	Indicates whether the primary or secondary data source should be used.
<b>Secondary Data Source</b>	The name of the secondary data source. This field is available only when <i>secondary</i> has been selected for the data source mode. This data source is used if the primary data source or the data item in the primary data source cannot be located.
<b>Allow QBE</b>	Indicates whether row-level record locking is selected for the data source. Row-level record locking should be selected to help prevent database integrity issues.

### See Also

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, “Working with Object Configuration Manager”*

*PeopleSoft EnterpriseOne Tools 8.95 PeopleBook: Configurable Network Computing Implementation, “Setting Up Data Sources”*



# Glossary of PeopleSoft Terms

<b>activity</b>	A scheduling entity in PeopleSoft EnterpriseOne Form Design Aid that represents a designated amount of time on a calendar.
<b>activity rule</b>	The criteria by which an object progresses from one given point to the next in a flow.
<b>add mode</b>	A condition of a form that enables users to input data.
<b>Advanced Planning Agent (APAg)</b>	A PeopleSoft EnterpriseOne tool that can be used to extract, transform, and load enterprise data. APAg supports access to data sources in the form of relational databases, flat file format, and other data or message encoding, such as XML.
<b>application server</b>	A server in a local area network that contains applications shared by network clients.
<b>as if processing</b>	A process that enables you to view currency amounts as if they were entered in a currency different from the domestic and foreign currency of the transaction.
<b>alternate currency</b>	<p>A currency that is different from the domestic currency (when dealing with a domestic-only transaction) or the domestic and foreign currency of a transaction.</p> <p>In PeopleSoft EnterpriseOne Financial Management, alternate currency processing enables you to enter receipts and payments in a currency other than the one in which they were issued.</p>
<b>as of processing</b>	A process that is run as of a specific point in time to summarize transactions up to that date. For example, you can run various PeopleSoft EnterpriseOne reports as of a specific date to determine balances and amounts of accounts, units, and so on as of that date.
<b>back-to-back process</b>	A process in PeopleSoft EnterpriseOne Workflow Management that contains the same keys that are used in another process.
<b>batch processing</b>	<p>A process of transferring records from a third-party system to PeopleSoft EnterpriseOne.</p> <p>In PeopleSoft EnterpriseOne Financial Management, batch processing enables you to transfer invoices and vouchers that are entered in a system other than EnterpriseOne to PeopleSoft EnterpriseOne Accounts Receivable and PeopleSoft EnterpriseOne Accounts Payable, respectively. In addition, you can transfer address book information, including customer and supplier records, to PeopleSoft EnterpriseOne.</p>
<b>batch server</b>	A server that is designated for running batch processing requests. A batch server typically does not contain a database nor does it run interactive applications.
<b>batch-of-one immediate</b>	<p>A transaction method that enables a client application to perform work on a client workstation, then submit the work all at once to a server application for further processing. As a batch process is running on the server, the client application can continue performing other tasks.</p> <p>See also direct connect and store-and-forward.</p>
<b>business function</b>	A named set of user-created, reusable business rules and logs that can be called through event rules. Business functions can run a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the application programming interfaces (APIs) that enable them to be called from a form, a database trigger, or a non-EnterpriseOne application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through

	event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.
<b>business function event rule</b>	See named event rule (NER).
<b>business view</b>	A means for selecting specific columns from one or more PeopleSoft EnterpriseOne tables whose data is used in an application or report. A business view does not select specific rows, nor does it contain any actual data. It is strictly a view through which you can manipulate data.
<b>central objects merge</b>	A process that blends a customer's modifications to the objects in a current release with objects in a new release.
<b>central server</b>	A server that has been designated to contain the originally installed version of the software (central objects) for deployment to client computers. In a typical PeopleSoft EnterpriseOne installation, the software is loaded on to one machine—the central server. Then, copies of the software are pushed out or downloaded to various workstations attached to it. That way, if the software is altered or corrupted through its use on workstations, an original set of objects (central objects) is always available on the central server.
<b>charts</b>	Tables of information in PeopleSoft EnterpriseOne that appear on forms in the software.
<b>connector</b>	Component-based interoperability model that enables third-party applications and PeopleSoft EnterpriseOne to share logic and data. The PeopleSoft EnterpriseOne connector architecture includes Java and COM connectors.
<b>contra/clearing account</b>	A general ledger account in PeopleSoft EnterpriseOne Financial Management that is used by the system to offset (balance) journal entries. For example, you can use a contra/clearing account to balance the entries created by allocations in PeopleSoft EnterpriseOne General Accounting.
<b>Control Table Workbench</b>	An application that, during the installation Workbench processing, runs the batch applications for the planned merges that update the data dictionary, user-defined codes, menus, and user override tables.
<b>control tables merge</b>	A process that blends a customer's modifications to the control tables with the data that accompanies a new release.
<b>cost assignment</b>	The process in PeopleSoft EnterpriseOne Advanced Cost Accounting of tracing or allocating resources to activities or cost objects.
<b>cost component</b>	In PeopleSoft EnterpriseOne Manufacturing Management, an element of an item's cost (for example, material, labor, or overhead).
<b>cross segment edit</b>	A logic statement that establishes the relationship between configured item segments. Cross segment edits are used to prevent ordering of configurations that cannot be produced.
<b>currency restatement</b>	The process of converting amounts from one currency into another currency, generally for reporting purposes. You can use the currency restatement process, for example, when many currencies must be restated into a single currency for consolidated reporting.
<b>database server</b>	A server in a local area network that maintains a database and performs searches for client computers.
<b>Data Source Workbench</b>	An application that, during the Installation Workbench process, copies all data sources that are defined in the installation plan from the Data Source Master and Table and Data Source Sizing tables in the Planner data source to the System-release number data source. It also updates the Data Source Plan detail record to reflect completion.

<b>date pattern</b>	A calendar that represents the beginning date for the fiscal year and the ending date for each period in that year in standard and 52-period accounting.
<b>denominated-in currency</b>	The company currency in which financial reports are based.
<b>deployment server</b>	A server that is used to install, maintain, and distribute software to one or more enterprise servers and client workstations.
<b>detail information</b>	Information that relates to individual lines in PeopleSoft EnterpriseOne transactions (for example, voucher pay items and sales order detail lines).
<b>direct connect</b>	A transaction method in which a client application communicates interactively and directly with a server application.  See also batch-of-one immediate and store-and-forward.
<b>Do Not Translate (DNT)</b>	A type of data source that must exist on the iSeries because of BLOB restrictions.
<b>dual pricing</b>	The process of providing prices for goods and services in two currencies.
<b>edit code</b>	A code that indicates how a specific value for a report or a form should appear or be formatted. The default edit codes that pertain to reporting require particular attention because they account for a substantial amount of information.
<b>edit mode</b>	A condition of a form that enables users to change data.
<b>edit rule</b>	A method used for formatting and validating user entries against a predefined rule or set of rules.
<b>Electronic Data Interchange (EDI)</b>	An interoperability model that enables paperless computer-to-computer exchange of business transactions between PeopleSoft EnterpriseOne and third-party systems. Companies that use EDI must have translator software to convert data from the EDI standard format to the formats of their computer systems.
<b>embedded event rule</b>	An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with the business function event rule.
<b>Employee Work Center</b>	A central location for sending and receiving all PeopleSoft EnterpriseOne messages (system and user generated), regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages.
<b>enterprise server</b>	A server that contains the database and the logic for PeopleSoft EnterpriseOne or PeopleSoft World.
<b>EnterpriseOne object</b>	A reusable piece of code that is used to build applications. Object types include tables, forms, business functions, data dictionary items, batch processes, business views, event rules, versions, data structures, and media objects.
<b>EnterpriseOne process</b>	A software process that enables PeopleSoft EnterpriseOne clients and servers to handle processing requests and run transactions. A client runs one process, and servers can have multiple instances of a process. PeopleSoft EnterpriseOne processes can also be dedicated to specific tasks (for example, workflow messages and data replication) to ensure that critical processes don't have to wait if the server is particularly busy.
<b>Environment Workbench</b>	An application that, during the Installation Workbench process, copies the environment information and Object Configuration Manager tables for each environment from the Planner data source to the System-release number data source. It also updates the Environment Plan detail record to reflect completion.
<b>escalation monitor</b>	A batch process that monitors pending requests or activities and restarts or forwards them to the next step or user after they have been inactive for a specified amount of time.

<b>event rule</b>	A logic statement that instructs the system to perform one or more operations based on an activity that can occur in a specific application, such as entering a form or exiting a field.
<b>facility</b>	An entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. A facility is sometimes referred to as a <i>business unit</i> .
<b>fast path</b>	A command prompt that enables the user to move quickly among menus and applications by using specific commands.
<b>file server</b>	A server that stores files to be accessed by other computers on the network. Unlike a disk server, which appears to the user as a remote disk drive, a file server is a sophisticated device that not only stores files, but also manages them and maintains order as network user request files and make changes to these files.
<b>final mode</b>	The report processing mode of a processing mode of a program that updates or creates data records.
<b>FTP server</b>	A server that responds to requests for files via file transfer protocol.
<b>header information</b>	Information at the beginning of a table or form. Header information is used to identify or provide control information for the group of records that follows.
<b>interface table</b>	See Z table.
<b>integration server</b>	A server that facilitates interaction between diverse operating systems and applications across internal and external networked computer systems.
<b>integrity test</b>	A process used to supplement a company's internal balancing procedures by locating and reporting balancing problems and data inconsistencies.
<b>interoperability model</b>	A method for third-party systems to connect to or access PeopleSoft EnterpriseOne.
<b>in-your-face-error</b>	In PeopleSoft EnterpriseOne, a form-level property which, when enabled, causes the text of application errors to appear on the form.
<b>IServer service</b>	Developed by PeopleSoft, this internet server service resides on the web server and is used to speed up delivery of the Java class files from the database to the client.
<b>jargon</b>	An alternative data dictionary item description that PeopleSoft EnterpriseOne or People World displays based on the product code of the current object.
<b>Java application server</b>	A component-based server that resides in the middle-tier of a server-centric architecture. This server provides middleware services for security and state maintenance, along with data access and persistence.
<b>JDBNET</b>	A database driver that enables heterogeneous servers to access each other's data.
<b>JDEBASE Database Middleware</b>	A PeopleSoft proprietary database middleware package that provides platform-independent APIs, along with client-to-server access.
<b>JDECallObject</b>	An API used by business functions to invoke other business functions.
<b>jde.ini</b>	A PeopleSoft file (or member for iSeries) that provides the runtime settings required for EnterpriseOne initialization. Specific versions of the file or member must reside on every machine running PeopleSoft EnterpriseOne. This includes workstations and servers.
<b>JDEIPC</b>	Communications programming tools used by server code to regulate access to the same data in multiprocess environments, communicate and coordinate between processes, and create new processes.

<b>jde.log</b>	The main diagnostic log file of PeopleSoft EnterpriseOne. This file is always located in the root directory on the primary drive and contains status and error messages from the startup and operation of PeopleSoft EnterpriseOne.
<b>JDENET</b>	PeopleSoft proprietary communications middleware package. This package is a peer-to-peer, message-based, socket-based, multiprocess communications middleware solution. It handles client-to-server and server-to-server communications for all PeopleSoft EnterpriseOne supported platforms.
<b>Location Workbench</b>	An application that, during the Installation Workbench process, copies all locations that are defined in the installation plan from the Location Master table in the Planner data source to the System data source.
<b>logic server</b>	A server in a distributed network that provides the business logic for an application program. In a typical configuration, pristine objects are replicated on to the logic server from the central server. The logic server, in conjunction with workstations, actually performs the processing required when PeopleSoft EnterpriseOne and World software runs.
<b>MailMerge Workbench</b>	An application that merges Microsoft Word 6.0 (or higher) word-processing documents with PeopleSoft EnterpriseOne records to automatically print business documents. You can use MailMerge Workbench to print documents, such as form letters about verification of employment.
<b>master business function (MBF)</b>	An interactive master file that serves as a central location for adding, changing, and updating information in a database. Master business functions pass information between data entry forms and the appropriate tables. These master functions provide a common set of functions that contain all of the necessary default and editing rules for related programs. MBFs contain logic that ensures the integrity of adding, updating, and deleting information from databases.
<b>master table</b>	See published table.
<b>matching document</b>	A document associated with an original document to complete or change a transaction. For example, in PeopleSoft EnterpriseOne Financial Management, a receipt is the matching document of an invoice, and a payment is the matching document of a voucher.
<b>media storage object</b>	Files that use one of the following naming conventions that are not organized into table format: Gxxx, xxxGT, or GTxxx.
<b>message center</b>	A central location for sending and receiving all PeopleSoft EnterpriseOne messages (system and user generated), regardless of the originating application or user.
<b>messaging adapter</b>	An interoperability model that enables third-party systems to connect to PeopleSoft EnterpriseOne to exchange information through the use of messaging queues.
<b>messaging server</b>	A server that handles messages that are sent for use by other programs using a messaging API. Messaging servers typically employ a middleware program to perform their functions.
<b>named event rule (NER)</b>	Encapsulated, reusable business logic created using event rules, rather than C programming. NERs are also called business function event rules. NERs can be reused in multiple places by multiple programs. This modularity lends itself to streamlining, reusability of code, and less work.
<b><i>nota fiscal</i></b>	In Brazil, a legal document that must accompany all commercial transactions for tax purposes and that must contain information required by tax regulations.
<b><i>nota fiscal factura</i></b>	In Brazil, a nota fiscal with invoice information. See also <i>nota fiscal</i> .

<b>Object Configuration Manager (OCM)</b>	In PeopleSoft EnterpriseOne, the object request broker and control center for the runtime environment. OCM keeps track of the runtime locations for business functions, data, and batch applications. When one of these objects is called, OCM directs access to it using defaults and overrides for a given environment and user.
<b>Object Librarian</b>	A repository of all versions, applications, and business functions reusable in building applications. Object Librarian provides check-out and check-in capabilities for developers, and it controls the creation, modification, and use of PeopleSoft EnterpriseOne objects. Object Librarian supports multiple environments (such as production and development) and enables objects to be easily moved from one environment to another.
<b>Object Librarian merge</b>	A process that blends any modifications to the Object Librarian in a previous release into the Object Librarian in a new release.
<b>Open Data Access (ODA)</b>	An interoperability model that enables you to use SQL statements to extract PeopleSoft EnterpriseOne data for summarization and report generation.
<b>Output Stream Access (OSA)</b>	An interoperability model that enables you to set up an interface for PeopleSoft EnterpriseOne to pass data to another software package, such as Microsoft Excel, for processing.
<b>package</b>	EnterpriseOne objects are installed to workstations in packages from the deployment server. A package can be compared to a bill of material or kit that indicates the necessary objects for that workstation and where on the deployment server the installation program can find them. It is point-in-time snap shot of the central objects on the deployment server.
<b>package build</b>	A software application that facilitates the deployment of software changes and new applications to existing users. Additionally, in PeopleSoft EnterpriseOne, a package build can be a compiled version of the software. When you upgrade your version of the ERP software, for example, you are said to take a package build.  Consider the following context: “Also, do not transfer business functions into the production path code until you are ready to deploy, because a global build of business functions done during a package build will automatically include the new functions.” The process of creating a package build is often referred to, as it is in this example, simply as “a package build.”
<b>package location</b>	The directory structure location for the package and its set of replicated objects. This is usually \\deployment server\release\path_code\package\package name. The subdirectories under this path are where the replicated objects for the package are placed. This is also referred to as where the package is built or stored.
<b>Package Workbench</b>	An application that, during the Installation Workbench process, transfers the package information tables from the Planner data source to the System-release number data source. It also updates the Package Plan detail record to reflect completion.
<b>PeopleSoft Database</b>	See JDEBASE Database Middleware.
<b>planning family</b>	A means of grouping end items whose similarity of design and manufacture facilitates being planned in aggregate.
<b>preference profile</b>	The ability to define default values for specified fields for a user-defined hierarchy of items, item groups, customers, and customer groups.
<b>print server</b>	The interface between a printer and a network that enables network clients to connect to the printer and send their print jobs to it. A print server can be a computer, separate hardware device, or even hardware that resides inside of the printer itself.
<b>pristine environment</b>	A PeopleSoft EnterpriseOne environment used to test unaltered objects with PeopleSoft demonstration data or for training classes. You must have this environment so that you can compare pristine objects that you modify.

<b>processing option</b>	A data structure that enables users to supply parameters that regulate the running of a batch program or report. For example, you can use processing options to specify default values for certain fields, to determine how information appears or is printed, to specify date ranges, to supply runtime values that regulate program execution, and so on.
<b>production environment</b>	A PeopleSoft EnterpriseOne environment in which users operate EnterpriseOne software.
<b>production-grade file server</b>	A file server that has been quality assurance tested and commercialized and that is usually provided in conjunction with user support services.
<b>program temporary fix (PTF)</b>	A representation of changes to PeopleSoft software that your organization receives on magnetic tapes or disks.
<b>project</b>	In PeopleSoft EnterpriseOne, a virtual container for objects being developed in Object Management Workbench.
<b>promotion path</b>	<p>The designated path for advancing objects or projects in a workflow. The following is the normal promotion cycle (path):</p> <p>11&gt;21&gt;26&gt;28&gt;38&gt;01</p> <p>In this path, <i>11</i> equals new project pending review, <i>21</i> equals programming, <i>26</i> equals QA test/review, <i>28</i> equals QA test/review complete, <i>38</i> equals in production, <i>01</i> equals complete. During the normal project promotion cycle, developers check objects out of and into the development path code and then promote them to the prototype path code. The objects are then moved to the productions path code before declaring them complete.</p>
<b>proxy server</b>	A server that acts as a barrier between a workstation and the internet so that the enterprise can ensure security, administrative control, and caching service.
<b>published table</b>	Also called a master table, this is the central copy to be replicated to other machines. Residing on the publisher machine, the F98DRPUB table identifies all of the published tables and their associated publishers in the enterprise.
<b>publisher</b>	The server that is responsible for the published table. The F98DRPUB table identifies all of the published tables and their associated publishers in the enterprise.
<b>pull replication</b>	One of the PeopleSoft methods for replicating data to individual workstations. Such machines are set up as pull subscribers using PeopleSoft EnterpriseOne data replication tools. The only time that pull subscribers are notified of changes, updates, and deletions is when they request such information. The request is in the form of a message that is sent, usually at startup, from the pull subscriber to the server machine that stores the F98DRPCN table.
<b>QBE</b>	An abbreviation for query by example. In PeopleSoft EnterpriseOne, the QBE line is the top line on a detail area that is used for filtering data.
<b>real-time event</b>	A service that uses system calls to capture PeopleSoft EnterpriseOne transactions as they occur and to provide notification to third-party software, end users, and other PeopleSoft systems that have requested notification when certain transactions occur.
<b>refresh</b>	A function used to modify PeopleSoft EnterpriseOne software, or subset of it, such as a table or business data, so that it functions at a new release or cumulative update level, such as B73.2 or B73.2.1.
<b>replication server</b>	A server that is responsible for replicating central objects to client machines.
<b>quote order</b>	In PeopleSoft EnterpriseOne Procurement and Subcontract Management, a request from a supplier for item and price information from which you can create a purchase order.

	In PeopleSoft EnterpriseOne Sales Order Management, item and price information for a customer who has not yet committed to a sales order.
<b>selection</b>	Found on PeopleSoft menus, a selection represents functions that you can access from a menu. To make a selection, type the associated number in the Selection field and press Enter.
<b>Server Workbench</b>	An application that, during the Installation Workbench process, copies the server configuration files from the Planner data source to the System-release number data source. It also updates the Server Plan detail record to reflect completion.
<b>spot rate</b>	An exchange rate entered at the transaction level. This rate overrides the exchange rate that is set up between two currencies.
<b>Specification merge</b>	A merge that comprises three merges: Object Librarian merge, Versions List merge, and Central Objects merge. The merges blend customer modifications with data that accompanies a new release.
<b>specification</b>	A complete description of a PeopleSoft EnterpriseOne object. Each object has its own specification, or name, which is used to build applications.
<b>Specification Table Merge Workbench</b>	An application that, during the Installation Workbench process, runs the batch applications that update the specification tables.
<b>store-and-forward</b>	The mode of processing that enables users who are disconnected from a server to enter transactions and then later connect to the server to upload those transactions.
<b>subscriber table</b>	Table F98DRSUB, which is stored on the publisher server with the F98DRPUB table and identifies all of the subscriber machines for each published table.
<b>supplemental data</b>	<p>Any type of information that is not maintained in a master file. Supplemental data is usually additional information about employees, applicants, requisitions, and jobs (such as an employee's job skills, degrees, or foreign languages spoken). You can track virtually any type of information that your organization needs.</p> <p>For example, in addition to the data in the standard master tables (the Address Book Master, Customer Master, and Supplier Master tables), you can maintain other kinds of data in separate, generic databases. These generic databases enable a standard approach to entering and maintaining supplemental data across PeopleSoft EnterpriseOne systems.</p>
<b>table access management (TAM)</b>	The PeopleSoft EnterpriseOne component that handles the storage and retrieval of use-defined data. TAM stores information, such as data dictionary definitions; application and report specifications; event rules; table definitions; business function input parameters and library information; and data structure definitions for running applications, reports, and business functions.
<b>Table Conversion Workbench</b>	An interoperability model that enables the exchange of information between PeopleSoft EnterpriseOne and third-party systems using non-PeopleSoft EnterpriseOne tables.
<b>table conversion</b>	An interoperability model that enables the exchange of information between PeopleSoft EnterpriseOne and third-party systems using non-PeopleSoft EnterpriseOne tables.
<b>table event rules</b>	Logic that is attached to database triggers that runs whenever the action specified by the trigger occurs against the table. Although PeopleSoft EnterpriseOne enables event rules to be attached to application events, this functionality is application specific. Table event rules provide embedded logic at the table level.
<b>terminal server</b>	A server that enables terminals, microcomputers, and other devices to connect to a network or host computer or to devices attached to that particular computer.

<b>three-tier processing</b>	The task of entering, reviewing and approving, and posting batches of transactions in PeopleSoft EnterpriseOne.
<b>three-way voucher match</b>	In PeopleSoft EnterpriseOne Procurement and Subcontract Management, the process of comparing receipt information to supplier's invoices to create vouchers. In a three-way match, you use the receipt records to create vouchers.
<b>transaction processing (TP) monitor</b>	A monitor that controls data transfer between local and remote terminals and the applications that originated them. TP monitors also protect data integrity in the distributed environment and may include programs that validate data and format terminal screens.
<b>transaction set</b>	An electronic business transaction (electronic data interchange standard document) made up of segments.
<b>trigger</b>	One of several events specific to data dictionary items. You can attach logic to a data dictionary item that the system processes automatically when the event occurs.
<b>triggering event</b>	A specific workflow event that requires special action or has defined consequences or resulting actions.
<b>two-way voucher match</b>	In PeopleSoft EnterpriseOne Procurement and Subcontract Management, the process of comparing purchase order detail lines to the suppliers' invoices to create vouchers. You do not record receipt information.
<b>User Overrides merge</b>	Adds new user override records into a customer's user override table.
<b>variance</b>	In Capital Asset Management, the difference between revenue generated by a piece of equipment and costs incurred by the equipment.  In EnterpriseOne Project Costing and EnterpriseOne Manufacturing Management, the difference between two methods of costing the same item (for example, the difference between the frozen standard cost and the current cost is an engineering variance). Frozen standard costs come from the Cost Components table, and the current costs are calculated using the current bill of material, routing, and overhead rates.
<b>Version List merge</b>	The Versions List merge preserves any non-XJDE and non-ZJDE version specifications for objects that are valid in the new release, as well as their processing options data.
<b>visual assist</b>	Forms that can be invoked from a control via a trigger to assist the user in determining what data belongs in the control.
<b>vocabulary override</b>	An alternate description for a data dictionary item that appears on a specific PeopleSoft EnterpriseOne or World form or report.
<b>wchar_t</b>	An internal type of a wide character. It is used for writing portable programs for international markets.
<b>web application server</b>	A web server that enables web applications to exchange data with the back-end systems and databases used in eBusiness transactions.
<b>web server</b>	A server that sends information as requested by a browser, using the TCP/IP set of protocols. A web server can do more than just coordination of requests from browsers; it can do anything a normal server can do, such as house applications or data. Any computer can be turned into a web server by installing server software and connecting the machine to the internet.
<b>Windows terminal server</b>	A multiuser server that enables terminals and minimally configured computers to display Windows applications even if they are not capable of running Windows software themselves. All client processing is performed centrally at the Windows terminal server and only display, keystroke, and mouse commands are transmitted over the network to the client terminal device.

<b>workbench</b>	A program that enables users to access a group of related programs from a single entry point. Typically, the programs that you access from a workbench are used to complete a large business process. For example, you use the EnterpriseOne Payroll Cycle Workbench (P07210) to access all of the programs that the system uses to process payroll, print payments, create payroll reports, create journal entries, and update payroll history. Examples of PeopleSoft EnterpriseOne workbenches include Service Management Workbench (P90CD020), Line Scheduling Workbench (P3153), Planning Workbench (P13700), Auditor's Workbench (P09E115), and Payroll Cycle Workbench.
<b>work day calendar</b>	In EnterpriseOne Manufacturing Management, a calendar that is used in planning functions that consecutively lists only working days so that component and work order scheduling can be done based on the actual number of work days available. A work day calendar is sometimes referred to as planning calendar, manufacturing calendar, or shop floor calendar.
<b>workflow</b>	The automation of a business process, in whole or in part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.
<b>workgroup server</b>	A server that usually contains subsets of data replicated from a master network server. A workgroup server does not perform application or batch processing.
<b>XAPI events</b>	A service that uses system calls to capture PeopleSoft EnterpriseOne transactions as they occur and then calls third-party software, end users, and other PeopleSoft systems that have requested notification when the specified transactions occur to return a response.
<b>XML CallObject</b>	An interoperability capability that enables you to call business functions.
<b>XML Dispatch</b>	An interoperability capability that provides a single point of entry for all XML documents coming into PeopleSoft EnterpriseOne for responses.
<b>XML List</b>	An interoperability capability that enables you to request and receive PeopleSoft EnterpriseOne database information in chunks.
<b>XML Service</b>	An interoperability capability that enables you to request events from one PeopleSoft EnterpriseOne system and receive a response from another PeopleSoft EnterpriseOne system.
<b>XML Transaction</b>	An interoperability capability that enables you to use a predefined transaction type to send information to or request information from PeopleSoft EnterpriseOne. XML transaction uses interface table functionality.
<b>XML Transaction Service (XTS)</b>	Transforms an XML document that is not in the PeopleSoft EnterpriseOne format into an XML document that can be processed by PeopleSoft EnterpriseOne. XTS then transforms the response back to the request originator XML format.
<b>Z event</b>	A service that uses interface table functionality to capture PeopleSoft EnterpriseOne transactions and provide notification to third-party software, end users, and other PeopleSoft systems that have requested to be notified when certain transactions occur.
<b>Z table</b>	A working table where non-PeopleSoft EnterpriseOne information can be stored and then processed into PeopleSoft EnterpriseOne. Z tables also can be used to retrieve PeopleSoft EnterpriseOne data. Z tables are also known as interface tables.
<b>Z transaction</b>	Third-party data that is properly formatted in interface tables for updating to the PeopleSoft EnterpriseOne database.

# Index

## A

- Add EnterpriseOne Object to the Project form 13, 55, 68
- Add Object form 13, 55, 68
- additional documentation xii
- Advanced Inputs form 75
- application fundamentals xi
- Assignment\Expression, description for using in event rules 27
- audit trail information, adding to tables 8

## B

- Batch Application Design form 68
- batch applications, searching in Cross Reference Facility 34
- Batch Delete
  - deleting records using 95
  - example 86
  - understanding 86
- batch versions
  - overriding properties 100
  - processing table conversions 99
- buffered inserts
  - getting error messages for 21
  - understanding 20
  - using in table I/O 24
  - using special operations 24
- business functions
  - description for using in event rules 27
- business functions, searching in Cross Reference Facility 35
- Business View Design Aid form 55, 56
- Business View Design Aid, forms included in 50
- business view joins, types 45
- business views
  - adding 55
  - creating table unions in 57
  - deleting components 51
  - designing 49
  - joining tables in 57
  - naming 49
  - naming conventions for external developers 50
  - naming joined business views 50

- searching in Cross Reference Facility 35
- selecting data items for 56
- selecting tables for 56
- understanding 45
- understanding primary key fields 48
- understanding Select Distinct 47
- understanding selecting data items for 51
- understanding selecting tables for 51
- understanding table joins 45
- understanding using Select Distinct 52
- using in table conversions 62
- using Select Distinct 56

## C

- comments, submitting xvi
- common elements xvii
- contact information xvi
- Copy Environment form 110, 111
- Copy Table form 13, 17
- CopyTableDataSource system function 96
- CopyTableEnvironment system function 96
- Cross Reference Facility
  - rebuilding cross reference information 37, 40
  - searching for batch applications 34
  - searching for business functions 35
  - searching for business views 35
  - searching for data items 34
  - searching for data structures 36
  - searching for event rules 36
  - searching for forms 36
  - searching for interactive applications 34
  - searching for objects 33
  - searching for tables 36
  - understanding 33
  - viewing field relationships 37
- Cross Reference form 38, 40
- cross reference information, rebuilding 37, 40
- cross-references xv

currency columns, using Select Distinct with 52  
 Customer Connection website xii

**D**

data  
   copying using Data Copy 83  
   copying using Data Copy with Table Input 85  
   understanding converting 69  
   understanding copying 82  
   understanding copying with table input 84  
   viewing in tables 12  
   viewing in tables using Universal Table Browser 18  
 Data Browser, viewing data in tables 12  
 Data Changed event, description 67  
 data conversion  
   example 70  
   understanding 69  
 Data Copy with Table Input, copying data using 85  
 Data Copy, copying data using 83  
 data dictionary, creating handles 24  
 data items  
   searching in Cross Reference Facility 34  
   selecting for business views 51, 56  
   using to create tables 9  
 data selection  
   defining in table conversions 72  
   understanding in table conversions 63  
 data sequencing  
   defining in table conversions 71  
   understanding in table conversions 63  
 Data Sequencing form 71  
 Data Source Revisions form 108  
 data sources, adding for foreign tables 108  
 data structures  
   attaching to table conversions 70  
   searching in Cross Reference Facility 36  
 database, understanding 5  
 debug logging  
   defining trace levels for 104  
   difference from logging options 104  
   setting the trace level on the server 105  
   setting the trace level on the workstation 105

default OCM mapping, setting up for foreign tables 111  
 DeleteCurrentInputRow 96  
 documentation  
   printed xii  
   related xii  
   updates xii

**E**

Else, description for using in event rules 28  
 environments, adding for foreign tables 110  
 ER Search form 38, 39  
 error messaging, using for buffered inserts 21  
 event flow in table conversions 65  
 event rules  
   creating in table conversions 97  
   creating table I/O 19, 23  
   events available for tables 27  
   functions available for tables 27  
   searching in Cross Reference Facility 36, 39  
   understanding in table conversions 96  
   understanding table I/O 19  
 Event Rules Design form 23, 97  
 events  
   order in which they occur in table conversions 66  
   used in tables 27  
 exporting text files 77  
 External Data form 70  
 external data, defining for table conversions 70  
 external developer considerations, naming conventions for business views 50

**F**

Field Relationships form 38, 39  
 field relationships, viewing in Cross Reference Facility 37, 39  
 file formats  
   defining for table conversions 76  
   defining multiple format delimited files 79  
   defining multiple format fixed-width files 81

- defining single format delimited files 78
- defining single format fixed-width files 80
- understanding user-defined 77
- Finish form 76
- fixed-width multiple format files, defining 81
- fixed-width single format files, defining 80
- flat files 61
- Flush Insert Buffer, using to maintain integrity of table I/O 21
- foreign tables
  - adding a PeopleSoft EnterpriseOne environment for 110
  - adding data sources for 108
  - definition of 62
  - setting up a default OCM mapping for 111
  - understanding 107
- Format Fetched event, description 67
- forms, searching in Cross Reference Facility 36

**G**

- Generate Indexes form 13, 16
- Generate Table form 13, 16
- generation
  - generating tables 11
  - understanding generating header files 11
  - understanding generating indices 11
- Get Error Data, using to retrieve errors for table I/O 21

**H**

- handles
  - creating 24
  - understanding 22
  - understanding using 22
  - using with table I/O 26
- header files
  - generating 17
  - understanding generating 11

**I**

- If\While, description for using in event rules 27

- importing text files 77
- indices
  - defining in tables 15
  - generating 16
  - naming 9
  - understanding 10
  - understanding generating 11
  - understanding guidelines 8
- inner joins, understanding in business views 45
- input environments
  - defining for table conversions 71
  - understanding in table conversions 63
- inputs
  - defining for table conversions 71
  - mapping to outputs in table conversions 74
- interactive applications, searching in Cross Reference Facility 34
- Introduction form 70, 83, 85, 86, 95

**J**

- jde.ini
  - defining the number of rows to process 105
  - forcing row-by-row conversion 105
  - reviewing the logging definition 100
  - reviewing the StopAfterRow setting 102
  - reviewing the Trace Level setting 102
  - setting the trace level for debug logging 105
  - setting the trace level on the workstation 105
- joined business views, naming 50
- joins, types in business views 45

**K**

- key fields, understanding 48

**L**

- left outer joins
  - diagram 47
  - understanding in business views 45
- List of values form 73
- logging options
  - deference from debug logging 104
  - selecting for table conversions 75
- Logging Options form 75

**M**

Machine Search & Select (W986110D)  
form 112  
Machine Search & Select (W986115E)  
form 108  
Mapping form 74  
mapping operators, valid for table I/O 20  
MMA Partners xii  
multiple-format files, defining 79

**N**

naming conventions  
naming business views 49  
naming business views for external  
development 50  
naming indices 9  
naming joined table business views 50  
naming tables 8  
navigation assistant, understanding 67  
New Column Properties form 78, 80, 81  
notes xv

**O**

Object Management Workbench form 13,  
28, 55, 68  
Object Mapping Revisions form 112  
objects, searching in Cross Reference  
Facility 33, 38  
OCM mappings  
setting up for foreign tables 112  
output environments  
defining for table conversions 71  
understanding in table conversions 63  
outputs  
defining in table conversions 73  
mapping to inputs in table  
conversions 74

**P**

PeopleBooks  
ordering xii  
PeopleCode, typographical  
conventions xiv  
PeopleSoft application fundamentals xi  
PeopleSoft Data Sources form 101  
PeopleSoft EnterpriseOne naming  
conventions for business views 50  
prerequisites xi  
primary key fields, understanding 48

printed documentation xii  
Process Begin event, description 67  
Process End event, description 67  
process flow, understanding in table  
conversions 65  
processing location for table conversions,  
overriding 102  
processing options, attaching to table  
conversions 70  
properties  
overriding at runtime 101  
overriding for new table conversion  
versions 102  
overriding in batch versions of table  
conversions 100  
Properties form 101, 102

**R**

Range of values form 73  
records, deleting from tables 86  
related documentation xii  
Remove Table form 13, 17  
right outer joins  
diagram 46  
understanding in business views 45  
Row Fetched event, description 67  
row-by-row conversion, defining in the  
jde.ini 105  
rows to process, defining in the jde.ini 105

**S**

Select Actions form 83  
Select Distinct  
example 53  
understanding 47  
understanding using 52  
using in business views 56  
using with currency columns 52  
Select Environments form 71  
Select Input form 71, 78, 79, 80, 81  
Select Object form 79, 80, 81, 82  
Select Output form 73, 78, 79, 80, 81  
server, setting the trace level in the  
jde.ini 105  
SetSelectionAppendFlag system  
function 97  
SetUserSelection system function 97  
simple joins  
diagram 46

- understanding in business views 45
- Single value form 73
- single-format files, defining 78
- special operations
  - example of Get Error Data 21
  - understanding Flush Insert Buffer and Get Error Data 21
- SQL 92 left outer joins, understanding in business views 46
- StopAfterRow
  - adding to the jde.ini 105
  - overriding the jde.ini 101, 102
- Submit Job-Work With Batch Versions-Available Versions form 103
- suggestions, submitting xvi
- system functions
  - CopyTableDataSource 96
  - CopyTableEnvironment 96
  - DeleteCurrentInputRow 96
  - description for using in event rules 28
  - SetSelectionAppendFlag 97
  - SetUserSelection 97
  - TCInsertRow 96
  - UpdateCurrentInputRow 96
  - UserInsertRow 96
  - using in table conversions 96

## T

- Table Conversion Design form 69
- Table Conversion Director 83, 85, 86, 95
- table conversion events
  - order in which they occur 66
  - understanding the flow 65
- table conversion system functions, descriptions 96
- Table Conversion Version Design form 102
- table conversions
  - adding object 68
  - adding table event rules 28
  - attaching data structures to 70
  - attaching processing options to 70
  - creating event rules in 97
  - defining external data 70
  - defining file formats for input or output 76
  - defining input 71
  - defining input and output environments 71
  - defining output 73

- defining sort and selection criteria 63
- deleting records using Batch Delete 95
- mapping inputs to outputs 74
- overriding properties at runtime 101
- overriding properties for new versions 102
- overriding the processing location 102
- reviewing results 76
- running on the web client 103
- selecting logging options 75
- starting the Director 69
- submitting 99
- submitting on the web client 100
- types of tables to use 62
- understanding Batch Delete 86
- understanding copying data 82
- understanding Data Copy with Table Input 84
- understanding input and output environments 63
- understanding processing 99
- understanding setup 65
- understanding testing 103
- understanding types of 61
- using business views in 62
- using text files in 62
- Table Design Aid
  - forms included in 9
- Table Design Aid form 13, 15, 17
- table event rule triggers, creating 28
- table event rules
  - adding to table conversions 28
  - creating triggers 28
  - functions available 27
  - understanding 27
- table I/O
  - creating in event rules 19, 23
  - describing operations available 19
  - describing valid mapping operators 20
  - understanding 19
  - understanding special operations 21
  - using buffered inserts 24
  - using handles in 26
- table I/O, description for using in event rules 28
- table joins
  - creating in business views 57
  - types for business views 45
  - understanding 54
  - understanding unions 47

Table Options form 85, 95  
 table unions  
   creating in business views 57  
   diagram 47  
   understanding 54  
   understanding in business views 47  
 tables  
   adding audit trail information 8  
   adding object 13  
   copying 17  
   creating using data items 9  
   generating 11, 16  
   modifying object information 15  
   naming 8  
   previewing 15  
   removing 17  
   reviewing information for 10  
   searching in Cross Reference Facility 36  
   selecting data items for 15  
   selecting for business views 51, 56  
   understanding converting data 69  
   understanding copying 11  
   understanding creation 7  
   understanding event rules in 27  
   understanding foreign 62  
   understanding indices guidelines 8  
   understanding removing 11  
   viewing data in 12  
     *See Also* foreign tables  
   viewing data using Universal Table Browser 18

TCInsertRow system function 96

text files  
   importing and exporting 77  
   using in table conversions 62

trace levels  
   defining for debug logging 104  
   defining in the jde.ini 105  
 triggers, adding to table event rules 28  
 typographical conventions xiv

## U

Universal Table Browser form 13, 18  
 Universal Table Browser, viewing data in tables 12  
 UpdateCurrentInputRow system function 96  
 User Defined Format - Column Delimiter form 78, 79, 81

User Defined Format - Column Layout form 78, 80, 81  
 User Defined Format - Finished form 79, 80, 81, 82  
 User Defined Format - Multiple Format Definition form 79, 81  
 User Defined Format - Multiple Format Names form 80, 81  
 User Defined Format - Type form 78, 79, 80, 81  
 user defined formats  
   defining multiple format fixed-width files 81  
   defining single format fixed-width files 80  
   understanding 77  
   using as input 77  
   using as output 78  
 UserInsertRow system function 96

## V

variables, description for using in event rules 28  
 visual cues xv

## W

warnings xv  
 web client, submitting table conversions 100  
 Work With Batch Versions-Available Versions form 101, 102  
 Work With Data Sources form 108  
 Work With Environments form 110  
 Work With Object Mappings form 112  
 workstation, setting the trace level 105