

JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Structure Design Guide

October 2007

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 is a registered trademark 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 JD Edwards 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 Documentation Prefacev**
- JD Edwards EnterpriseOne Application Prerequisites.....v
- Application Fundamentals.....v
- Documentation Updates and Printed Documentation.....vi
 - Obtaining Documentation Updates.....vi
 - Downloading Documentation.....vi
- Additional Resources.....vi
- Typographical Conventions and Visual Cues.....vii
 - Typographical Conventions.....viii
 - Visual Cues.....viii
 - Country, Region, and Industry Identifiers.....ix
 - Currency Codes.....x
- Comments and Suggestions.....x
- Common Fields Used in Implementation Guides.....x

Preface

- JD Edwards EnterpriseOne Data Structure Design Preface.....xiii**
- JD Edwards EnterpriseOne Tools Fundamentals.....xiii

Chapter 1

- Getting Started with JD Edwards EnterpriseOne Data Structure Design.....1**
- Data Structure Design Overview.....1
- Data Structure Design Implementation.....1
 - Data Structure Design Implementation Steps.....1

Chapter 2

- Understanding Data Structures.....3**
- Data Structures.....3
 - System-Generated Data Structures.....3
 - User-Generated Data Structures.....3

Chapter 3

Using Processing Options.....5

Understanding Processing Options.....5

Understanding Processing Options Templates.....5

Defining a Processing Options Data Structure (Template).....6

 Prerequisite.....6

 JD Edwards EnterpriseOne Processing Option Naming Standards.....7

 Language Considerations for Processing Options.....8

Attaching a Processing Options Template.....8

Working with Processing Options.....9

 Forms Used to Work with Processing Options.....9

 Defining a Processing Options Data Structure (Template).....9

 Changing a Template for Text Translation.....10

 Attaching a Processing Options Template.....10

Chapter 4

Creating Data Structures.....13

Creating Business Function Data Structures.....13

 Forms Used to Create a Data Structure.....13

 Creating Business Function Data Structures.....13

 Selecting Data Items for Business Function Data Structures.....14

 Creating a Type Definition.....15

Creating Processing Option Data Structures.....15

 Forms Used to Create a Processing Option Data Structure.....16

 Creating Processing Option Data Structures.....16

 Selecting Data Items for Processing Option Data Structures.....17

Creating Media Object Data Structures.....18

 Forms Used to Create Media Object Data Structures.....18

 Creating Media Object Data Structures.....18

 Selecting Data Items for Media Object Data Structures.....19

Glossary of JD Edwards EnterpriseOne Terms.....21

Index37

About This Documentation Preface

JD Edwards EnterpriseOne implementation guides provide you with the information that you need to implement and use JD Edwards EnterpriseOne applications from Oracle.

This preface discusses:

- JD Edwards EnterpriseOne application prerequisites.
- Application fundamentals.
- Documentation updates and printed documentation.
- Additional resources.
- Typographical conventions and visual cues.
- Comments and suggestions.
- Common fields in implementation guides.

Note. Implementation guides document only elements, such as fields and check boxes, that require additional explanation. If an 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 fields for the section, chapter, implementation guide, or product line. Fields that are common to all JD Edwards EnterpriseOne applications are defined in this preface.

JD Edwards EnterpriseOne Application Prerequisites

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

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

You should be familiar with navigating the system and adding, updating, and deleting information by using JD Edwards EnterpriseOne menus, 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 JD Edwards EnterpriseOne applications most effectively.

Application Fundamentals

Each application implementation guide provides implementation and processing information for your JD Edwards EnterpriseOne applications.

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 implementation guide. Most product lines have a version of the application fundamentals implementation guide. The preface of each implementation guide identifies the application fundamentals implementation guides that are associated with that implementation guide.

The application fundamentals implementation guide consists of important topics that apply to many or all JD Edwards EnterpriseOne applications. 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 implementation guides. They provide the starting points for fundamental implementation tasks.

Documentation Updates and Printed Documentation

This section discusses how to:

- Obtain documentation updates.
- Download documentation.

Obtaining Documentation Updates

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

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

See Also

Oracle's PeopleSoft Customer Connection, http://www.oracle.com/support/support_peoplesoft.html

Downloading Documentation

In addition to the complete line of documentation that is delivered on your implementation guide CD-ROM, Oracle makes JD Edwards EnterpriseOne documentation available to you via Oracle's website. You can download PDF versions of JD Edwards EnterpriseOne documentation online via the Oracle Technology Network. Oracle makes these PDF files available online for each major release shortly after the software is shipped.

See Oracle Technology Network, <http://www.oracle.com/technology/documentation/psftent.html>.

Additional Resources

The following resources are located on Oracle's PeopleSoft Customer Connection website:

Resource	Navigation
Application maintenance information	Updates + Fixes
Business process diagrams	Support, Documentation, Business Process Maps

Resource	Navigation
Interactive Services Repository	Support, Documentation, Interactive Services Repository
Hardware and software requirements	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Hardware and Software Requirements
Installation guides	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Installation Guides and Notes
Integration information	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Pre-Built Integrations for PeopleSoft Enterprise and JD Edwards EnterpriseOne Applications
Minimum technical requirements (MTRs)	Implement, Optimize + Upgrade; Implementation Guide; Supported Platforms
Documentation updates	Support, Documentation, Documentation Updates
Implementation guides support policy	Support, Support Policy
Prerelease notes	Support, Documentation, Documentation Updates, Category, Release Notes
Product release roadmap	Support, Roadmaps + Schedules
Release notes	Support, Documentation, Documentation Updates, Category, Release Notes
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 implementation guides:

Typographical Convention or Visual Cue	Description
Bold	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 JD Edwards EnterpriseOne or other book-length publication titles. In PeopleCode syntax, italic items are placeholders for arguments that your program must supply. We also use italics when we refer to words as words or letters as letters, as in the following: Enter the letter <i>O</i> .
KEY+KEY	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press the W key.
Monospace font	Indicates a PeopleCode program or other code example.
“ ” (quotation marks)	Indicate chapter titles in cross-references and words that are used differently from their intended meanings.
. . . (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe ().
[] (square brackets)	Indicate optional items in PeopleCode syntax.
& (ampersand)	When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object. Ampersands also precede all PeopleCode variables.

Visual Cues

Implementation guides contain the following visual cues.

Notes

Notes indicate information that you should pay particular attention to as you work with the JD Edwards EnterpriseOne 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

Implementation guides 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 implementation guides:

- 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 implementation guides:

- 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 and suggestions are important to us. We encourage you to send us your feedback about our PeopleBooks and other reference and training materials. Please include the release numbers for the PeopleTools and applications that you are currently using. Email your comments to PSOFT-INFODEV_US@ORACLE.COM.

Common Fields Used in Implementation Guides

Address Book Number	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.
As If Currency Code	Enter the three-character code to specify the currency that you want to use to view transaction amounts. This code enables 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.
Batch Number	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).
Batch Date	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.
Batch Status	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 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 <i>E</i> .

U: 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.

Branch/Plant	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.
Business Unit	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.
Category Code	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.
Company	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.
Currency Code	Enter the three-character code that represents the currency of the transaction. JD Edwards EnterpriseOne provides currency codes that are recognized by the International Organization for Standardization (ISO). The system stores currency codes in the F0013 table.
Document Company	<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>
Document Number	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.
Document Type	<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. JD Edwards 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>
Effective Date	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:

- The date on which a change of address becomes effective.
- The date on which a lease becomes effective.
- The date on which a price becomes effective.
- The date on which the currency exchange rate becomes effective.
- The date on which a tax rate becomes effective.

Fiscal Period and Fiscal Year

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).

G/L Date (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.

JD Edwards EnterpriseOne Data Structure Design Preface

This preface discusses Oracle's JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Structure Design.

JD Edwards EnterpriseOne Tools Fundamentals

Additional information describing the setup and design of the JD Edwards EnterpriseOne Tools system resides in companion documentation that apply to many or all JD Edwards EnterpriseOne Tools. These companion guides contain information that applies specifically to JD Edwards EnterpriseOne Tools Data Structure Design:

- *JD Edwards EnterpriseOne Tools 8.97 System Administration Guide*
- *JD Edwards EnterpriseOne Tools 8.97 JD Edwards Object Management Workbench Guide*

See Also

JD Edwards EnterpriseOne Tools 8.97 System Administration Guide, “Getting Started with JD Edwards EnterpriseOne Tools System Administration”

JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide, “Getting Started with JD Edwards EnterpriseOne OMW”

CHAPTER 1

Getting Started with JD Edwards EnterpriseOne Data Structure Design

This chapter discusses:

- Data Structure Design overview.
- Data Structure Design implementation.

Data Structure Design Overview

Oracle's JD Edwards Data Structure Design is used to create and modify JD Edwards EnterpriseOne data structures. Data structures are composed of data items defined in the data dictionary and are used to pass data to and from interactive and batch applications.

Data Structure Design Implementation

This section provides an overview of the steps that are required to implement Data Structure Design.

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

Data Structure Design Implementation Steps

This table lists the steps for the Development Tools: Data Structure Design implementation.

Step	Reference
1. Configure JD Edwards Object Management Workbench.	<i>JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide</i> , "Configuring JD Edwards EnterpriseOne OMW"
2. Configure JD Edwards Object Management Workbench user roles and allowed actions.	<i>JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide</i> , "Configuring User Roles and Allowed Actions"
3. Configure JD Edwards Object Management Workbench functions.	<i>JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide</i> , "Configuring JD Edwards EnterpriseOne OMW Functions"

Step	Reference
4. Configure JD Edwards Object Management Workbench activity rules.	<i>JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide</i> , “Configuring Activity Rules”
5. Configure JD Edwards Object Management Workbench save locations.	<i>JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide</i> , “Configuring Object Save Locations”
6. Set up default location and printers.	<i>JD Edwards EnterpriseOne Tools 8.97 Development Tools: Report Printing Administration Technologies Guide</i> , “Getting Started with JD Edwards EnterpriseOne Report Printing Administration Technologies”

CHAPTER 2

Understanding Data Structures

This chapter discusses data structures.

Data Structures

Data structures are a key element of any programming language. A data structure is a collection of data items used to pass data to other components of the same application or to another application entirely. Data can also be passed from a user to an interactive or batch application. The Oracle's JD Edwards EnterpriseOne objects that use data structures are forms, reports, and business functions. Data structures pass data among objects and applications to aid in the execution of JD Edwards EnterpriseOne applications. Data structures are either system-generated or created by the application developer.

System-Generated Data Structures

There are two types of system-generated data structures:

- Form Data Structure
- Report Data Structure

Form Data Structures

Each form of an interactive application contains a form data structure. This data structure is used to pass values to and from the form. You can modify the form data structure during development of the application. If you need to modify the data structure after development is complete, verify that any data items you wish to remove are not used in the form's event rules. You must redefine the affected event rules if you modify fields included in the form data structure. Form data structures are modified in Form Design Aid.

Report Data Structures

A report is a batch application. A batch application can receive values from a data structure or write values to a data structure. By default, the system creates an empty data structure for a batch application. You can add and delete fields for the data structure from the File menu in JD Edwards Report Design Aid.

User-Generated Data Structures

An application developer can create three types of data structures:

- Media object data structures
- Business function data structures
- Processing options data structures

Media Object Data Structures

Media objects are attachments to records. They store data related to the record attached. Media objects are stored in the F00165 table. The different types of media objects available are text, image, OLE (on Windows platform), shortcut, or URL file. You must select a data structure when attaching media objects. This data structure passes data between the application and the media object table. Typically, the data items included in this data structure are the primary key fields of the record. Media object data structures are developed using Data Structure Design.

Business Function Data Structures

A business function requires a data structure. The data structure defines values passed in and out of the business function. Business function data structures are developed using Data Structure Design. These data structures are created using data items defined in the data dictionary. They are attached to business functions which can be called from Event Rules. All business functions require a data structure, both C business functions and named event rule business functions. The data structure provides the parameters for passing data between the business function and the interactive or batch application. Modifying an existing data structure can have a significant impact on your system. Use the Cross Reference Facility to review all applications that will be affected by modifying the data structure.

Processing Options Data Structures

Processing options are used to provide initial values to the application or report when it is started. Processing Options can be used to customize an application or report. A processing option data structure defines data items that can be customized for the application or report. Processing option data structures are developed in Processing Option Design. Processing options can be displayed when a user launches an application or report, or they can be stored when a particular version is created.

CHAPTER 3

Using Processing Options

This chapter provides overviews of processing options and processing options templates, and discusses how to define a processing options data structure (template), attach a processing options template, and work with processing options.

Understanding Processing Options

Processing options are a set of start up values that are provided to an interactive or batch application when it is launched. You can use processing options to change the way in which an application or a report appears or behaves. You can attach different processing option values to different versions of the same application, which enables you to change the behavior of an application without creating a new application. Additionally, you can use processing options to:

- Control the path that a user can use to navigate through a system.
- Set up default values.
- Customize an application for different companies or different users.
- Control the format of forms and reports.
- Control page breaks and totaling for reports.
- Specify the default version of a related application or batch process.

Understanding Processing Options Templates

A processing option template is a special kind of data structure. It contains one or more data dictionary items. It can also contain one or more tab pages that you can use to categorize data items.

Each tab page has a descriptive title.

You can add data items on processing option templates. Each data item contains a descriptive label and an edit field.

You can also add comments on a tab page. A comment is a text string that is displayed on the processing option dialog. It helps users understand the data items.

At runtime, a processing option dialog displays a set of tabs. Each tab represents a category of processing options. When you click the tab header, the tab body displays the set of processing options for that category.

Important! Changes to processing option text can conflict with changes to processing option templates. Template changes do not take effect until another package is built, but text changes occur immediately.

The following steps describe how to create and implement processing options:

1. Create processing option template.
2. Attach this template to an application and create event rules so that the application uses these values.
3. Create versions of the application. Save different processing option values to different versions. Specify the default version.
4. Specify how the processing options are handled at application launch time. You can set up the menu to do one of the following:
 - The processing options dialog appears. The user can select to supply values to processing options.
 - A version list appears. The user can select a version to launch.
 - The system runs a particular version with the processing option values saved for that version.

At runtime, depending on how you set up the application, one of the following events occurs:

Processing option template is created through Oracle's JD Edwards Processing Option Design Aid. When working with processing option design aid (PODA), all processing option template information is stored in Processing Option Text (POTEXT) Table Access Management (TAM) specifications until you check it in. When you check in the processing option template, it is moved from POTEXT TAM to the F98306 table. Data values for processing options are stored in the F983051 table. For batch versions, the Versions List table has an identifier that points to specifications for overrides (report overrides, data sequencing, data selection, or override location).

Each version of an application can be associated with a list of processing option values.

Defining a Processing Options Data Structure (Template)

You can create a processing options data structure (template) that lists the values for data items that are passed to the application at runtime. Any changes that you make to the template reside on the workstation until you check in the template. This ensures that current users of the existing template are not immediately affected by the changes. After you check in the changes, the next Just-In-Time-Install (JITI) replicates the changes to the other users.

Prerequisite

Create a processing options data structure.

See [Chapter 4, "Creating Data Structures," Creating Processing Option Data Structures, page 15](#).

JD Edwards EnterpriseOne Processing Option Naming Standards

You should follow JD Edwards EnterpriseOne naming standards whenever possible unless you have a strong business case for doing otherwise. Following the naming standards ensures a consistent approach to programming.

Processing Option Data Structure

The name of a data structure can be a maximum of 10 characters-only 9 if you begin the name with a T and is formatted as T xxxxyyyy:

Where:

T = Processing option data structure.

xxxxyyy = The program number for the application or report.

For example, the processing option data structure name for the P0101 application is T0101.

Tab Title

Use these guidelines when you define a tab title for a processing option:

- Avoid abbreviations.
- For future processing options, indicate that they are currently unavailable by entering the word FUTURE. If the entire tab is unavailable, enter FUTURE behind the extended description for the tab. If a single processing option is unavailable, place FUTURE behind the data item description.
- Ensure that each tab exists only one time and that it is not divided into multiple tabs. For example, use Process instead of Process 1, Process 2.
- Include the application name, such as P4310, in the text when referencing versions that are to be used. The Version tab should always begin with the comment block. Enter the version to be used for each program. If left blank, ZJDE0001 will be used.
- Use application-specific tabs sparingly and only when no other categories are appropriate. To allow for increased length of text when the text is translated, the name of an application-specific tab should be no longer than 10 characters in English.
- Use one of the eight standard tab titles. Along with the extended description and processing options for each, they are as follows:
- Display: Options that determine whether specific fields appear or which format of a form appears on entry.
- Defaults: Options that assign default values to specific fields.
- Edits: Options that indicate whether the system performs data validation for specific fields.
- Process: Options that control the process flow of the application.

Application-specific tabs are:

- Currency: Options that are specific to currency.
- Categories: Options that assign default category codes.
- Print: Options that control the output of a report.

- Versions: Options that specify which versions the system runs of applications that are called from this application.

Comment

Use these guidelines when you enter a comment for a processing option:

Note. When several processing options are grouped together, you can select to number the processing options or the comments. Choose whatever works best for the situation.

- Number every option on a tab. Use sequential numbering, starting at 1, for each tab.
- Use nouns, such as Customer Master, to describe the processing option. The action required is defined in the glossary for that processing option.
- Add the word "Required" to the end of the processing option if a processing option is required.
- Use a comment block when multiple processing options refer to the same topic. The comment block is a title for the logical group of processing options.

Data Item

Use these guidelines when creating a data item for a processing option:

- When necessary, change the name of the data item to be descriptive.
- When renaming the data item element, the field element should comply with the naming standards for event rule variables, with the alias appended, such as szCategoryCode3_CT03.
- Use a relevant data item when the data dictionary glossary applies. The user can display the glossary from the processing options. Do not use generic work fields, such as EV01.

Language Considerations for Processing Options

You can change a processing option template to incorporate language features.

When you add a new processing option template for an application that is language-enabled, complete the following tasks:

1. Create the application.
2. Create the processing option template for the base language.
3. Add the language text.

Attaching a Processing Options Template

You must attach a processing options template (data structure) to an application to enable processing options at runtime. A processing options template has these characteristics:

- It exists as a separate object.
- It can be attached to multiple applications.

When you attach a processing options template, if any of the processing options are designed to process on a certain event, you must attach event rule logic to enable those processing options.

Working with Processing Options

This section discusses how to:

- Define a processing options data structure (template).
- Change a template for text translation.
- Attach a processing options template.

Forms Used to Work with Processing Options

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	Type <i>OMW</i> in the Fast Path field of Solution Explorer.	View and select objects in projects.
Work With PO Text Translations	W98306A	Type <i>P98306</i> in the Fast Path field of Solution Explorer.	Change a template for text translation.

Defining a Processing Options Data Structure (Template)

Access the JD Edwards Object Management Workbench form.

1. Check out the processing options data structure with which you want to work.
2. Ensure that the data structure is highlighted, and then click the Design button in the center column.
3. On Processing Option Design, click the Design Tools tab, and then click Start the Processing Option Design Aid.

The Processing Options Design tool launches. The area on the left of the form displays how the processing option will look to the user.

4. Locate the data items that you need for the processing options with the Data Dictionary Browser.
5. Use one of these methods to select the data items that you want to add to the processing options:
 - Double-click the item in the Data Dictionary Browser. The item appears in the left side of the form under the tab.
 - Drag the item from the Data Dictionary Browser to the position where you want it in the structure members.
6. Click an item to edit it.
 - You can use the hatching around the control to reposition it.

- You can select text, and then delete or overwrite it.

Processing Options Design automatically adjusts the size and position of data items to fit the width of the tab.

7. Click the text button (A) to add comments.
8. Choose an object in the area on the left side of the form, and select Properties from the View menu.
Right-click a data item to view its properties and change the item name if necessary. The item name should be unique.
You can click the Help Override Data Item tab to add an alternate data dictionary name from which to get the help.
9. Right-click the processing option, and then select Properties from the menu.
10. On Data Item Properties, click the Help Override Data Item tab, and then complete the Data Item Help Override Name field.

Note. When you name Help Override Data Items, you should use the naming guidelines as defined in the Development Guidelines for Application Design.

11. Click OK.
12. To view tab properties, click the tab and select Properties from the View menu.
You can also right-click a tab and select Current Tab Properties from the menu that appears.
On Current Tab Properties, you can enter a short and long name for the tab.
Use the Help File Name field to add the name of the help file for the tab.
13. To add a new tab, select New Tab from the File menu.
You can also right-click an existing tab and select New Tab from the menu that appears.

Changing a Template for Text Translation

Access the Work With PO Text Translations form.

1. Complete these fields and click Find:
 - Template Name
 - To Language

Work with PO Text Translations displays processing option text for the processing option template and language that you specify.
2. Choose a row with the text type that you want to change and click Select.
Text types include tabs, items, and comments.
3. On PO Text Translation, enter the new text.

Attaching a Processing Options Template

Access the Form Design Aid.

1. On JD Edwards Form Design Aid, from the file menu, select Application Properties.
2. On the Application tab, click the .. button under the Processing Options Template.
3. On Select Processing Option Template, select the processing option template that you want to use and click OK.

Important! If you disconnect that template from the application or connect a different template to the application, the application might not run properly.

To change the processing option template, first remove all existing versions of the application. Then, examine all event rules within the application to ensure that references to the old processing option items are removed. Then attach the new processing option template.

CHAPTER 4

Creating Data Structures

This chapter discusses how to:

- Create business function data structures.
- Create processing option data structures.
- Create media object data structures.

Creating Business Function Data Structures

When you create a business function, you must attach a data structure. The data structure is used to pass data between an interactive or batch application and the business function. These data structures are created and modified using Oracle's JD Edwards Data Structure Design accessed from JD Edwards Object Management Workbench. Business function data structures contain data items defined in the data dictionary.

This section discusses how to create business function data structures.

Forms Used to Create a Data Structure

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	In the Fast Path field of Solution Explorer, type <i>OMW</i> .	Add and manage objects.
Add EnterpriseOne Object to the Project	W98220C	Select a project on OMW and click the Add button.	Select a new object to add to a project.
Add Object	W9861AF	Select Data Structure on the Add Object form and click the OK button.	Enter the name, description, and product code of a data structure. For a regular data structure, select Regular Data Structure.
Data Structure Design	W9860A	Complete the Add Object form and select Regular Data Structure, then click the OK button.	Select the Design Tools tab, and then click the Data Structure Design button.

Creating Business Function Data Structures

Business function data structures are regular data structures used in creating business functions.

Access the Add Object form.

Object Name	Enter a unique name for the data structure. It needs to be unique within the entire JD Edwards EnterpriseOne system.
Product Code	Enter a user-defined code (98/SY) that identifies a system. Values include: <i>01</i> Address Book <i>03B</i> Accounts Receivable <i>04</i> Accounts Payable <i>09</i> General Accounting <i>11</i> Multicurrency
Product System Code	Enter a user-defined code (98/SY) that specifies the system number for reporting and jargon purposes.
Object Use	Enter a code that designates the use of the object. By default, data structure is 360. See UDC 98/FU.
Regular Data Structure	Select this option if this is a business function data structure.

Selecting Data Items for Business Function Data Structures

Data items for business function data structures are data structures used in creating business functions.

To select data items for business function data structures:

1. Click the Design Tools tab, and then click the Data Structure Design button.
2. On the Dictionary Items tab, use the QBE line to locate the data dictionary items that you want to include in the business function data structure.
3. To include data items in the data structure, drag them from the Dictionary Items tab to Structure Members.

You can view detailed information about data items after they have been moved to Structure Members. Click the data item and select Data Dictionary Detail from the toolbar. The Data Item Specifications form appear.

4. To remove data items from the data structure, select the data item in Structure Members and click the Delete button on the toolbar.
5. Optionally, you can define data items as required and indicate the direction the data will flow.

If the business function data structure will be used in a smart field, you must define the appropriate fields as required and set the direction arrows.

Selecting an X in the required field indicates that the fields are optional. A check mark indicates that the fields are required.

6. Optionally, you can add attachments to the data structure or to a data item in the data structure.
Click Data Structure Attachments on the toolbar to add attachments to the data structure.
Click a data item and then click Data Structure Item Attachments on the toolbar.
7. Click XREF on the toolbar to launch the Cross Reference Facility from Data Structure Design.
Use Cross Reference Facility when modifying existing data structures. Validate all references after the change.
8. When the data structure is complete, click OK.

Creating a Type Definition

When the business function data structure is complete, and you are ready to use it in a C business function, you need to create a type definition. A type definition is a C code representation of a data structure definition. The type definition is stored in the clipboard so that you can easily paste it into the appropriate section of the .h file. This process is more efficient because you do not have to type the code into the .h file and, therefore, it minimizes errors.

Access the Data Structure Design form.

1. Click the Design Tools tab.
2. Click the Create a type definition button.

You will see a message on the status bar that indicates that the type definition is stored in the clipboard. You can now paste it into the appropriate section of the .h file using the CTRL+ V key combination.

Creating Processing Option Data Structures

A processing option data structure is used to pass data to an interactive or batch application. Processing option data structures contain data items defined in the data dictionary. Processing option data structure is created through Processing Option Design Aid.

This section discusses how to create a processing options data structure.

Forms Used to Create a Processing Option Data Structure

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	In the Fast Path field of Solution Explorer, type <i>OMW</i> .	Add and manage objects.
Add EnterpriseOne Object to the Project	W98220C	Select a project on OMW and click the Add button.	Select a new object to add to a project.
Add Object	W9861AF	Select Data Structure on the Add Object form and click the OK button.	Enter the name, description, and product code of a data structure. For a processing option template, select Processing Option Template.
Processing Option Design	W9860A	Complete the Add Object form and select Processing Template, then click the OK button.	Select the Design Tools tab, and then click the Start Processing Option Design Aid button.

Creating Processing Option Data Structures

Processing options are used to prompt for values that are then passed to the interactive or batch application and are used to determine the data to be presented.

Access the Add Object form.

Object Name Enter a unique name for the data structure. It needs to be unique within the entire JD Edwards EnterpriseOne system.

Product Code Enter a user-defined code (98/SY) that identifies a system. Values include:

01

Address Book

03B

Accounts Receivable

04

Accounts Payable

09

General Accounting

11

Multicurrency

Product System Code Enter a user-defined code (98/SY) that specifies the system number for reporting and jargon purposes.

Object Use Enter a code that designates the use of the object. The default value for data structure is 360.

See UDC 98/FU.

Processing Option Template

Select this option when creating processing option data structures.

Selecting Data Items for Processing Option Data Structures

Data items for processing option data structures are used to determine the data to be presented.

To select data items for processing option data structures:

1. Click the Design Tools tab, and then click Start the Processing Option Design Aid.
2. Right-click the processing option tab, <New Tab>, and select Current Tab Properties.
3. In the Sort Name and Long Name fields, enter a name for the processing option tab according to the recommended naming conventions. Click OK.
4. Optionally, click the letter A on the toolbar and click the processing option template.
This option drops a Comment Text field on the processing option template. Double-click the field and highlight the text. Enter comments to explain the options.
5. On the Data Dictionary Browser form, use the QBE line to locate the data items that you want to include on the processing option template.
All of the data items included on a single tab should be related and the tab name should be descriptive. To add additional unrelated data items, you should create new tabs.
6. To include data items in the processing option, drag them from the Data Dictionary Browser form to the processing option template.
7. Optionally, double-click the data item description and enter descriptive text.
This is necessary when you select a generic data item such as Everest Event Point 01 (EV01). This data item is used often when a single character value is expected, such as 1 or 2. You can change the text to read: Enter 1 to print a single column of labels, enter 2 to print two columns of labels.
When you change the name of a data item on a processing option template, you should change the member name as well. Right-click the entry field of the data item. Select Properties. Using Hungarian Notation rules, change the member name. Leave the prefix that indicates the field type and modify the name to associate it with the description on the processing option template. Append an underline and the alias of the data item used for the field. For example, cLabels_EV01. Do not use spaces or special characters in the member name.
8. To remove data items from the processing option, select the data item on the processing option template and click Cut Item on the toolbar.
9. Optionally, right-click the processing option tab and select New Tab.
In the Sort Name and Long Name fields, enter a name for the processing option tab according to the recommended naming conventions. Add related data items to this tab using the previous instructions.
10. When all of the tabs and data items required have been added to the processing option template, select Test from the Edit menu.
This option enables you to view the processing option template as it will appear to the user. If any of the data items are defined to include a visual assist, you will be able to access it from this test template.
11. Save the processing option data structure.

The processing option is now ready to be attached to an interactive or batch application. In the application, you must create event rules that define to the system how each processing option value should be processed.

Creating Media Object Data Structures

This section discusses how to create media object data structures.

Forms Used to Create Media Object Data Structures

Form Name	FormID	Navigation	Usage
Object Management Workbench	W98220A	Enter <i>OMW</i> in the Fast Path field.	View and manage objects in projects.
Add EnterpriseOne Object to the Project	W98220C	From JD Edwards Object Management Workbench, select a project, and click the Add button.	Select the type of object to add to the project.
Add Object	W9861AF	On Add JD Edwards EnterpriseOne Object to the project, select Media Object Data Structure and click the OK button.	Enter information about the media object data structure.
Data Structure Design	W9860AL	Complete the Add Object form and click the OK button.	Select the Design Tools tab, and then click the Data Structure Design button to add data items into the data structure.

Creating Media Object Data Structures

Media Object data structures are data structures used to access media object table records.

Access the Add Object form.

Object Name Enter a unique name for the data structure. It needs to be unique within the entire JD Edwards EnterpriseOne system.

Product Code Enter a user-defined code (98/SY) that identifies a system. Values include:

01

Address Book

03B

Accounts Receivable

04

	Accounts Payable
	<i>09</i>
	General Accounting
	<i>11</i>
	Multicurrency
Product System Code	Enter a user-defined code (98/SY) that specifies the system number for reporting and jargon purposes.
Object Use	Enter a code that designates the use of the object. The default value is “Undefined”.
	See UDC 98/FU.

Selecting Data Items for Media Object Data Structures

Media objects are attachments to records which are related to the record and are stored in the media object table.

To select data items for business function data structures:

1. Click the Design Tools tab, and then click the Data Structure Design button.
2. On the Dictionary Items tab, use the QBE line to locate the data dictionary items that you want to include in the media object data structure.
3. To include data items in the data structure, drag them from the Dictionary Items tab to Structure Members.
You can view detailed information about data items after they have been moved to Structure Members. Click the data item and select Data Dictionary Detail from the toolbar. The Data Item Specifications form appear.
4. To remove data items from the data structure, select the data item in Structure Members and click the Delete button on the toolbar.
5. Optionally, you can add attachments to the data structure or to a data item in the data structure.
Click Data Structure Attachments on the toolbar to add attachments to the data structure.
Click a data item and then click Data Structure Item Attachments on the toolbar.
6. Click XREF on the toolbar to launch the Cross Reference Facility from Data Structure Design.
Use Cross Reference Facility when modifying existing data structures. Validate all references after the change.
7. When the data structure is complete, click OK.

Glossary of JD Edwards EnterpriseOne Terms

Accessor Methods/Assessors	Java methods to “get” and “set” the elements of a value object or other source file.
activity rule	The criteria by which an object progresses from one given point to the next in a flow.
add mode	A condition of a form that enables users to input data.
Advanced Planning Agent (APAg)	A JD Edwards 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.
alternate currency	<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 JD Edwards 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>
Application Server	Software that provides the business logic for an application program in a distributed environment. The servers can be Oracle Application Server (OAS) or WebSphere Application Server (WAS).
as if processing	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.
as of processing	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 JD Edwards EnterpriseOne reports as of a specific date to determine balances and amounts of accounts, units, and so on as of that date.
Auto Commit Transaction	A database connection through which all database operations are immediately written to the database.
back-to-back process	A process in JD Edwards EnterpriseOne Supply Management that contains the same keys that are used in another process.
batch processing	<p>A process of transferring records from a third-party system to JD Edwards EnterpriseOne.</p> <p>In JD Edwards EnterpriseOne Financial Management, batch processing enables you to transfer invoices and vouchers that are entered in a system other than JD Edwards EnterpriseOne to JD Edwards EnterpriseOne Accounts Receivable and JD Edwards EnterpriseOne Accounts Payable, respectively. In addition, you can transfer address book information, including customer and supplier records, to JD Edwards EnterpriseOne.</p>
batch server	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.
batch-of-one immediate	<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>
best practices	Non-mandatory guidelines that help the developer make better design decisions.

BPEL	Abbreviation for Business Process Execution Language, a standard web services orchestration language, which enables you to assemble discrete services into an end-to-end process flow.
BPEL PM	Abbreviation for Business Process Execution Language Process Manager, a comprehensive infrastructure for creating, deploying, and managing BPEL business processes.
Build Configuration File	Configurable settings in a text file that are used by a build program to generate ANT scripts. ANT is a software tool used for automating build processes. These scripts build published business services.
build engineer	An actor that is responsible for building, mastering, and packaging artifacts. Some build engineers are responsible for building application artifacts, and some are responsible for building foundation artifacts.
Build Program	A WIN32 executable that reads build configuration files and generates an ANT script for building published business services.
business analyst	An actor that determines if and why an EnterpriseOne business service needs to be developed.
business function	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-JD Edwards 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.
business function event rule	See named event rule (NER).
business service	EnterpriseOne business logic written in Java. A business service is a collection of one or more artifacts. Unless specified otherwise, a business service implies both a published business service and business service.
business service artifacts	Source files, descriptors, and so on that are managed for business service development and are needed for the business service build process.
business service class method	A method that accesses resources provided by the business service framework.
business service configuration files	Configuration files include, but are not limited to, interop.ini, JDBj.ini, and jdelog.properties.
business service cross reference	A key and value data pair used during orchestration. Collectively refers to both the code and the key cross reference in the WSG/XPI based system.
business service cross-reference utilities	Utility services installed in a BPEL/ESB environment that are used to access JD Edwards EnterpriseOne orchestration cross-reference data.
business service development environment	A framework needed by an integration developer to develop and manage business services.
business services development tool	Otherwise known as JDeveloper.
business service EnterpriseOne object	A collection of artifacts managed by EnterpriseOne LCM tools. Named and represented within EnterpriseOne LCM similarly to other EnterpriseOne objects like tables, views, forms, and so on.

business service framework	Parts of the business service foundation that are specifically for supporting business service development.
business service payload	An object that is passed between an enterprise server and a business services server. The business service payload contains the input to the business service when passed to the business services server. The business service payload contains the results from the business service when passed to the Enterprise Server. In the case of notifications, the return business service payload contains the acknowledgement.
business service property	Key value data pairs used to control the behavior or functionality of business services.
Business Service Property Admin Tool	An EnterpriseOne application for developers and administrators to manage business service property records.
business service property business service group	A classification for business service property at the business service level. This is generally a business service name. A business service level contains one or more business service property groups. Each business service property group may contain zero or more business service property records.
business service property categorization	A way to categorize business service properties. These properties are categorized by business service.
business service property key	A unique name that identifies the business service property globally in the system.
business service property utilities	A utility API used in business service development to access EnterpriseOne business service property data.
business service property value	A value for a business service property.
business service repository	A source management system, for example ClearCase, where business service artifacts and build files are stored. Or, a physical directory in network.
business services server	The physical machine where the business services are located. Business services are run on an application server instance.
business services source file or business service class	One type of business service artifact. A text file with the .java file type written to be compiled by a Java compiler.
business service value object template	The structural representation of a business service value object used in a C-business function.
Business Service Value Object Template Utility	A utility used to create a business service value object template from a business service value object.
business services server artifact	The object to be deployed to the business services server.
business view	A means for selecting specific columns from one or more JD Edwards EnterpriseOne application 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.
central objects merge	A process that blends a customer's modifications to the objects in a current release with objects in a new release.
central server	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 JD Edwards 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.

charts	Tables of information in JD Edwards EnterpriseOne that appear on forms in the software.
check-in repository	A repository for developers to check in and check out business service artifacts. There are multiple check-in repositories. Each can be used for a different purpose (for example, development, production, testing, and so on).
connector	Component-based interoperability model that enables third-party applications and JD Edwards EnterpriseOne to share logic and data. The JD Edwards EnterpriseOne connector architecture includes Java and COM connectors.
contra/clearing account	A general ledger account in JD Edwards 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 JD Edwards EnterpriseOne Financial Management.
Control Table Workbench	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.
control tables merge	A process that blends a customer's modifications to the control tables with the data that accompanies a new release.
correlation data	The data used to tie HTTP responses with requests that consist of business service name and method.
cost assignment	The process in JD Edwards EnterpriseOne Advanced Cost Accounting of tracing or allocating resources to activities or cost objects.
cost component	In JD Edwards EnterpriseOne Manufacturing, an element of an item's cost (for example, material, labor, or overhead).
credentials	A valid set of JD Edwards EnterpriseOne username/password/environment/role, EnterpriseOne session, or EnterpriseOne token.
Cross-reference utility services	Utility services installed in a BPEL/ESB environment that access EnterpriseOne cross-reference data.
cross segment edit	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.
currency restatement	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.
cXML	A protocol used to facilitate communication between business documents and procurement applications, and between e-commerce hubs and suppliers.
database credentials	A valid database username/password.
database server	A server in a local area network that maintains a database and performs searches for client computers.
Data Source Workbench	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.
date pattern	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.

denominated-in currency	The company currency in which financial reports are based.
deployment artifacts	Artifacts that are needed for the deployment process, such as servers, ports, and such.
deployment server	A server that is used to install, maintain, and distribute software to one or more enterprise servers and client workstations.
detail information	Information that relates to individual lines in JD Edwards EnterpriseOne transactions (for example, voucher pay items and sales order detail lines).
direct connect	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.
Do Not Translate (DNT)	A type of data source that must exist on the iSeries because of BLOB restrictions.
dual pricing	The process of providing prices for goods and services in two currencies.
duplicate published business services authorization records	Two published business services authorization records with the same user identification information and published business services identification information.
embedded application server instance	An OC4J instance started by and running wholly within JDeveloper.
edit code	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.
edit mode	A condition of a form that enables users to change data.
edit rule	A method used for formatting and validating user entries against a predefined rule or set of rules.
Electronic Data Interchange (EDI)	An interoperability model that enables paperless computer-to-computer exchange of business transactions between JD Edwards 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.
embedded event rule	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.
Employee Work Center	A central location for sending and receiving all JD Edwards 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.
enterprise server	A server that contains the database and the logic for JD Edwards EnterpriseOne.
Enterprise Service Bus (ESB)	Middleware infrastructure products or technologies based on web services standards that enable a service-oriented architecture using an event-driven and XML-based messaging framework (the bus).
EnterpriseOne administrator	An actor responsible for the EnterpriseOne administration system.
EnterpriseOne credentials	A user ID, password, environment, and role used to validate a user of EnterpriseOne.
EnterpriseOne object	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.

EnterpriseOne development client	Historically called “fat client,” a collection of installed EnterpriseOne components required to develop EnterpriseOne artifacts, including the Microsoft Windows client and design tools.
EnterpriseOne extension	A JDeveloper component (plug-in) specific to EnterpriseOne. A JDeveloper wizard is a specific example of an extension.
EnterpriseOne process	A software process that enables JD Edwards 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. JD Edwards 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.
EnterpriseOne resource	Any EnterpriseOne table, metadata, business function, dictionary information, or other information restricted to authorized users.
Environment Workbench	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.
escalation monitor	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.
event rule	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.
explicit transaction	Transaction used by a business service developer to explicitly control the type (auto or manual) and the scope of transaction boundaries within a business service.
exposed method or value object	Published business service source files or parts of published business service source files that are part of the published interface. These are part of the contract with the customer.
facility	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 “business unit.”
fast path	A command prompt that enables the user to move quickly among menus and applications by using specific commands.
file server	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 users request files and make changes to these files.
final mode	The report processing mode of a processing mode of a program that updates or creates data records.
foundation	A framework that must be accessible for execution of business services at runtime. This includes, but is not limited to, the Java Connector and JDBj.
FTP server	A server that responds to requests for files via file transfer protocol.
header information	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.
HTTP Adapter	A generic set of services that are used to do the basic HTTP operations, such as GET, POST, PUT, DELETE, TRACE, HEAD, and OPTIONS with the provided URL.

instantiate	A Java term meaning “to create.” When a class is instantiated, a new instance is created.
integration developer	The user of the system who develops, runs, and debugs the EnterpriseOne business services. The integration developer uses the EnterpriseOne business services to develop these components.
integration point (IP)	The business logic in previous implementations of EnterpriseOne that exposes a document level interface. This type of logic used to be called XBPs. In EnterpriseOne 8.11, IPs are implemented in Web Services Gateway powered by webMethods.
integration server	A server that facilitates interaction between diverse operating systems and applications across internal and external networked computer systems.
integrity test	A process used to supplement a company’s internal balancing procedures by locating and reporting balancing problems and data inconsistencies.
interface table	See Z table.
internal method or value object	Business service source files or parts of business service source files that are not part of the published interface. These could be private or protected methods. These could be value objects not used in published methods.
interoperability model	A method for third-party systems to connect to or access JD Edwards EnterpriseOne.
in-your-face-error	In JD Edwards EnterpriseOne, a form-level property which, when enabled, causes the text of application errors to appear on the form.
iServer service	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.
jargon	An alternative data dictionary item description that JD Edwards EnterpriseOne appears based on the product code of the current object.
Java application server	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.
JDBNET	A database driver that enables heterogeneous servers to access each other’s data.
JDEBASE Database Middleware	A JD Edwards EnterpriseOne proprietary database middleware package that provides platform-independent APIs, along with client-to-server access.
JDECallObject	An API used by business functions to invoke other business functions.
jde.ini	A JD Edwards EnterpriseOne file (or member for iSeries) that provides the runtime settings required for JD Edwards EnterpriseOne initialization. Specific versions of the file or member must reside on every machine running JD Edwards EnterpriseOne. This includes workstations and servers.
JDEIPC	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.
jde.log	The main diagnostic log file of JD Edwards 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 JD Edwards EnterpriseOne.
JDENET	A JD Edwards EnterpriseOne 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 JD Edwards EnterpriseOne supported platforms.
JDeveloper Project	An artifact that JDeveloper uses to categorize and compile source files.

JDeveloper Workspace	An artifact that JDeveloper uses to organize project files. It contains one or more project files.
JMS Queue	A Java Messaging service queue used for point-to-point messaging.
listener service	A listener that listens for XML messages over HTTP.
local repository	A developer's local development environment that is used to store business service artifacts.
local standalone BPEL/ESB server	A standalone BPEL/ESB server that is not installed within an application server.
Location Workbench	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.
logic server	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 JD Edwards EnterpriseOne software runs.
MailMerge Workbench	An application that merges Microsoft Word 6.0 (or higher) word-processing documents with JD Edwards EnterpriseOne records to automatically print business documents. You can use MailMerge Workbench to print documents, such as form letters about verification of employment.
Manual Commit transaction	A database connection where all database operations delay writing to the database until a call to commit is made.
master business function (MBF)	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.
master table	See published table.
matching document	A document associated with an original document to complete or change a transaction. For example, in JD Edwards EnterpriseOne Financial Management, a receipt is the matching document of an invoice, and a payment is the matching document of a voucher.
media storage object	Files that use one of the following naming conventions that are not organized into table format: Gxxx, xxxGT, or GTxxx.
message center	A central location for sending and receiving all JD Edwards EnterpriseOne messages (system and user generated), regardless of the originating application or user.
messaging adapter	An interoperability model that enables third-party systems to connect to JD Edwards EnterpriseOne to exchange information through the use of messaging queues.
messaging server	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.
Middle-Tier BPEL/ESB Server	A BPEL/ESB server that is installed within an application server.
Monitoring Application	An EnterpriseOne tool provided for an administrator to get statistical information for various EnterpriseOne servers, reset statistics, and set notifications.

named event rule (NER)	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.
<i>nota fiscal</i>	In Brazil, a legal document that must accompany all commercial transactions for tax purposes and that must contain information required by tax regulations.
<i>nota fiscal factura</i>	In Brazil, a <i>nota fiscal</i> with invoice information. See also <i>nota fiscal</i> .
Object Configuration Manager (OCM)	In JD Edwards 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.
Object Librarian	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 JD Edwards EnterpriseOne objects. Object Librarian supports multiple environments (such as production and development) and enables objects to be easily moved from one environment to another.
Object Librarian merge	A process that blends any modifications to the Object Librarian in a previous release into the Object Librarian in a new release.
Open Data Access (ODA)	An interoperability model that enables you to use SQL statements to extract JD Edwards EnterpriseOne data for summarization and report generation.
Output Stream Access (OSA)	An interoperability model that enables you to set up an interface for JD Edwards EnterpriseOne to pass data to another software package, such as Microsoft Excel, for processing.
package	JD Edwards 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 snapshot of the central objects on the deployment server.
package build	A software application that facilitates the deployment of software changes and new applications to existing users. Additionally, in JD Edwards 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.”
package location	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.
Package Workbench	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.
Pathcode Directory	The specific portion of the file system on the EnterpriseOne development client where EnterpriseOne development artifacts are stored.

patterns	General repeatable solutions to a commonly occurring problem in software design. For business service development, the focus is on the object relationships and interactions. For orchestrations, the focus is on the integration patterns (for example, synchronous and asynchronous request/response, publish, notify, and receive/reply).
planning family	A means of grouping end items whose similarity of design and manufacture facilitates being planned in aggregate.
preference profile	The ability to define default values for specified fields for a user-defined hierarchy of items, item groups, customers, and customer groups.
print server	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.
pristine environment	A JD Edwards EnterpriseOne environment used to test unaltered objects with JD Edwards EnterpriseOne demonstration data or for training classes. You must have this environment so that you can compare pristine objects that you modify.
processing option	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.
production environment	A JD Edwards EnterpriseOne environment in which users operate EnterpriseOne software.
production-grade file server	A file server that has been quality assurance tested and commercialized and that is usually provided in conjunction with user support services.
Production Published Business Services Web Service	Published business services web service deployed to a production application server.
program temporary fix (PTF)	A representation of changes to JD Edwards EnterpriseOne software that your organization receives on magnetic tapes or disks.
project	In JD Edwards EnterpriseOne, a virtual container for objects being developed in Object Management Workbench.
promotion path	<p>The designated path for advancing objects or projects in a workflow. The following is the normal promotion cycle (path):</p> <p>11>21>26>28>38>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>
proxy server	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.
published business service	EnterpriseOne service level logic and interface. A classification of a published business service indicating the intention to be exposed to external (non-EnterpriseOne) systems.
published business service identification information	Information about a published business service used to determine relevant authorization records. Published business services + method name, published business services, or *ALL.

published business service web service	Published business services components packaged as J2EE Web Service (namely, a J2EE EAR file that contains business service classes, business service foundation, configuration files, and web service artifacts).
published table	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.
publisher	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.
pull replication	One of the JD Edwards EnterpriseOne methods for replicating data to individual workstations. Such machines are set up as pull subscribers using JD Edwards 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.
QBE	An abbreviation for query by example. In JD Edwards EnterpriseOne, the QBE line is the top line on a detail area that is used for filtering data.
real-time event	A message triggered from EnterpriseOne application logic that is intended for external systems to consume.
refresh	A function used to modify JD Edwards 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.
replication server	A server that is responsible for replicating central objects to client machines.
Rt-Addressing	Unique data identifying a browser session that initiates the business services call request host/port user session.
rules	Mandatory guidelines that are not enforced by tooling, but must be followed in order to accomplish the desired results and to meet specified standards.
quote order	In JD Edwards Procurement and Subcontract Management, a request from a supplier for item and price information from which you can create a purchase order. In JD Edwards Sales Order Management, item and price information for a customer who has not yet committed to a sales order.
secure by default	A security model that assumes that a user does not have permission to execute an object unless there is a specific record indicating such permissions.
Secure Socket Layer (SSL)	A security protocol that provides communication privacy. SSL enables client and server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery.
SEI implementation	A Java class that implements the methods that declare in a Service Endpoint Interface (SEI).
selection	Found on JD Edwards EnterpriseOne 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.
serialize	The process of converting an object or data into a format for storage or transmission across a network connection link with the ability to reconstruct the original data or objects when needed.
Server Workbench	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. The application also updates the Server Plan detail record to reflect completion.
Service Endpoint Interface (SEI)	A Java interface that declares the methods that a client can invoke on the service.
SOA	Abbreviation for Service Oriented Architecture.
soft coding	A coding technique that enables an administrator to manipulate site-specific variables that affect the execution of a given process.
source repository	A repository for HTTP adapter and listener service development environment artifacts.
spot rate	An exchange rate entered at the transaction level. This rate overrides the exchange rate that is set up between two currencies.
Specification merge	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.
specification	A complete description of a JD Edwards EnterpriseOne object. Each object has its own specification, or name, which is used to build applications.
Specification Table Merge Workbench	An application that, during the Installation Workbench process, runs the batch applications that update the specification tables.
SSL Certificate	A special message signed by a certificate authority that contains the name of a user and that user's public key in such a way that anyone can "verify" that the message was signed by no one other than the certification authority and thereby develop trust in the user's public key.
store-and-forward	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.
subscriber table	Table F98DRSUB, which is stored on the publisher server with the F98DRPUB table and identifies all of the subscriber machines for each published table.
superclass	An inheritance concept of the Java language where a class is an instance of something, but is also more specific. "Tree" might be the superclass of "Oak" and "Elm," for example.
supplemental data	<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 JD Edwards EnterpriseOne systems.</p>
table access management (TAM)	The JD Edwards 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.
Table Conversion Workbench	An interoperability model that enables the exchange of information between JD Edwards EnterpriseOne and third-party systems using non-JD Edwards EnterpriseOne tables.

table conversion	An interoperability model that enables the exchange of information between JD Edwards EnterpriseOne and third-party systems using non-JD Edwards EnterpriseOne tables.
table event rules	Logic that is attached to database triggers that runs whenever the action specified by the trigger occurs against the table. Although JD Edwards 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.
terminal server	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.
three-tier processing	The task of entering, reviewing and approving, and posting batches of transactions in JD Edwards EnterpriseOne.
three-way voucher match	In JD Edwards 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.
transaction processing (TP) monitor	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.
transaction processing method	A method related to the management of a manual commit transaction boundary (for example, start, commit, rollback, and cancel).
transaction set	An electronic business transaction (electronic data interchange standard document) made up of segments.
trigger	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.
triggering event	A specific workflow event that requires special action or has defined consequences or resulting actions.
two-way authentication	An authentication mechanism in which both client and server authenticate themselves by providing the SSL certificates to each other.
two-way voucher match	In JD Edwards 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.
user identification information	User ID, role, or *public.
User Overrides merge	Adds new user override records into a customer's user override table.
value object	A specific type of source file that holds input or output data, much like a data structure passes data. Value objects can be exposed (used in a published business service) or internal, and input or output. They are comprised of simple and complex elements and accessories to those elements.
variance	In JD Edwards Capital Asset Management, the difference between revenue generated by a piece of equipment and costs incurred by the equipment. In JD Edwards EnterpriseOne Project Costing and JD Edwards EnterpriseOne Manufacturing, 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.

versioning a published business service	Adding additional functionality/interfaces to the published business services without modifying the existing functionality/interfaces.
Version List merge	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.
visual assist	Forms that can be invoked from a control via a trigger to assist the user in determining what data belongs in the control.
vocabulary override	An alternate description for a data dictionary item that appears on a specific JD Edwards EnterpriseOne form or report.
wchar_t	An internal type of a wide character. It is used for writing portable programs for international markets.
web application server	A web server that enables web applications to exchange data with the back-end systems and databases used in eBusiness transactions.
web server	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.
Web Service Description Language (WSDL)	An XML format for describing network services.
Web Service Inspection Language (WSIL)	An XML format for assisting in the inspection of a site for available services and a set of rules for how inspection-related information should be made.
web service proxy foundation	Foundation classes for web service proxy that must be included in a business service server artifact for web service consumption on WAS.
web service softcoding record	An XML document that contains values that are used to configure a web service proxy. This document identifies the endpoint and conditionally includes security information.
web service softcoding template	An XML document that provides the structure for a soft coded record.
Where clause	The portion of a database operation that specifies which records the database operation will affect.
Windows terminal server	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.
wizard	A type of JDeveloper extension used to walk the user through a series of steps.
workbench	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 JD Edwards 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 JD Edwards EnterpriseOne workbenches include Service Management Workbench (P90CD020), Line Scheduling Workbench (P3153), Planning Workbench (P13700), Auditor's Workbench (P09E115), and Payroll Cycle Workbench.
work day calendar	In JD Edwards EnterpriseOne Manufacturing, 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.

workflow	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.
workgroup server	A server that usually contains subsets of data replicated from a master network server. A workgroup server does not perform application or batch processing.
XAPI events	A service that uses system calls to capture JD Edwards EnterpriseOne transactions as they occur and then calls third-party software, end users, and other JD Edwards EnterpriseOne systems that have requested notification when the specified transactions occur to return a response.
XML CallObject	An interoperability capability that enables you to call business functions.
XML Dispatch	An interoperability capability that provides a single point of entry for all XML documents coming into JD Edwards EnterpriseOne for responses.
XML List	An interoperability capability that enables you to request and receive JD Edwards EnterpriseOne database information in chunks.
XML Service	An interoperability capability that enables you to request events from one JD Edwards EnterpriseOne system and receive a response from another JD Edwards EnterpriseOne system.
XML Transaction	An interoperability capability that enables you to use a predefined transaction type to send information to or request information from JD Edwards EnterpriseOne. XML transaction uses interface table functionality.
XML Transaction Service (XTS)	Transforms an XML document that is not in the JD Edwards EnterpriseOne format into an XML document that can be processed by JD Edwards EnterpriseOne. XTS then transforms the response back to the request originator XML format.
Z event	A service that uses interface table functionality to capture JD Edwards EnterpriseOne transactions and provide notification to third-party software, end users, and other JD Edwards EnterpriseOne systems that have requested to be notified when certain transactions occur.
Z table	A working table where non-JD Edwards EnterpriseOne information can be stored and then processed into JD Edwards EnterpriseOne. Z tables also can be used to retrieve JD Edwards EnterpriseOne data. Z tables are also known as interface tables.
Z transaction	Third-party data that is properly formatted in interface tables for updating to the JD Edwards EnterpriseOne database.

Index

A

- add JD Edwards EnterpriseOne object to the project form 13, 16, 18
- add object form 13, 16, 18
- additional documentation vi
- additional features
 - processing options 5
- application fundamentals v

B

- business function data structures
 - creating 13
 - overview 4

C

- comments 8
- comments, submitting x
- common fields x
- contact information x
- creating a processing options data structure 6
- creating data structures 13, 15
- creating media object data structures 18
- cross-references ix
- Customer Connection website vi

D

- data item
 - selecting a data item for a processing option 8
- data structure design form 13, 14, 15, 18, 19
- data structures
 - business function 4
 - creating 6, 13, 15, 18
 - form 3
 - fundamentals 3
 - media object 4
 - processing option 4
 - report 3
 - system generated 3
 - type definition 15
 - user generated 4
- documentation

- printed vi
- related vi
- updates vi

F

- form data structures 3
- Form Design Aid 10
- forms
 - add JD Edwards EnterpriseOne object to the project 13, 16, 18
 - add object 13, 16, 18
 - data structure design 13, 14, 15, 18, 19
 - object management workbench 13, 16, 18
 - Object Management Workbench 9
 - processing option design 16
 - processing options design 17
 - Work With PO Text Translations 9, 10
- fundamentals
 - data structures 3

I

- implementation guides
 - ordering vi

L

- language considerations for processing options 8

M

- media object data structures
 - creating 18
 - overview 4

N

- notes ix

O

- object management workbench form 13, 16, 18
- Object Management Workbench form 9
- Object Management Workbench forms 9
- object name 14, 16, 18
- object use 14, 16, 19

P

- PeopleCode, typographical conventions viii
- prerequisites v
- printed documentation vi
- processing option data structures
 - creating 15
 - overview 4
- processing option design form 16, 17
- processing option template 17
- processing option template, changing for text translation 10
- processing options
 - adding a processing template with translated text 8
 - attaching a processing options template 8
 - batch application 5
 - data structure 6, 7
 - interactive application 5
 - language considerations 8
 - naming standards 7, 8
 - templates 5
- product code 14, 16, 18
- product system code 14, 16, 19

R

- regular data structure 14
- related documentation vi
- report data structures 3

S

- suggestions, submitting x
- system generated data structures
 - form data structures 3
 - report data structures 3

T

- tab title 7
- templates, processing options 5
- translated text 8
- typographical conventions viii

U

- user generated data structures
 - business function data structures 4
 - media object data structures 4
 - processing options data structures 4

V

- visual cues viii

W

- warnings ix
- Work With PO Text Translations form 9, 10