ORACLE
JD EDWARDS ENTERPRISEONE

JD Edwards EnterpriseOne Tools 8.97 Development Tools: Overview

October 2007

ORACLE

JD Edwards EnterpriseOne Tools 8.97 Development Tools: Overview
SKU E1_TOOLSTDT-B1007

Contents

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.
Italics	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 *O*.
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:

Blank: Batch is unposted and pending approval.

A: The batch is approved for posting, has no errors and is in balance, but has not yet been posted.

D: The batch posted successfully.

E: The batch is in error. You must correct the batch before it can post.

P: The system is in the process of posting the batch. The batch is unavailable until the posting process is complete. If errors occur during the post, the batch status changes to *E*.

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 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.

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.

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.

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 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: Accounts payable documents.

R: Accounts receivable documents.

T: Time and pay documents.

I: Inventory documents.

O: Purchase order documents.

S: Sales order documents.

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 Development Tools: Overview Preface

This preface discusses JD Edwards Development Tools: Overview.

JD Edwards EnterpriseOne Development Tools

Oracle's JD Edwards EnterpriseOne Development Tools are an integrated set of application development tools. These tools allow business analysts to develop complete interactive and batch applications, such as forms and reports. The tools simplify the development process and limit the amount of programming necessary to create applications. JD EnterpriseOne Development Tools also allow you to use the stability of JD EnterpriseOne methodology and the ease of the Microsoft Windows interface to create applications for client/server environments.

CHAPTER 1

Understanding JD Edwards EnterpriseOne Acronyms

This chapter discusses the acronyms that are commonly used JD Edwards EnterpriseOne Development Tools.

Acronym List

Many of the acronyms that are commonly used in EnterpriseOne Development Tools are defined in the following table:

Term	Description
BDA	Business View Design Aid
BSFN	Business Function
BSVW	Business View
CSV	Comma Separated Values
DD	Data Dictionary
DS or DSTR	Data Structure
ER	Event Rules
FDA	Form Design Aid
H4A	HTML for Applications
NER	Named Event Rule
OCM	Object Configuration Manager
OL	Object Librarian
OMC	Object Management Configuration
OMW	Object Management Workbench
OSA	Output Stream Access

Term	Description
PO	Processing Option
QBE	Query by Example
RDA	Report Design Aid
SAR	Software Action Request
TAM	Table Access Management
TBLE	Table
TC	Table Conversions
TDA	Table Design Aid
TER	Table Event Rule
UBE	Universal Batch Engine
UDC	User Defined Code
UTB	Universal Table Browser
WF	Workflow
XREF	Cross Reference Facility

CHAPTER 2

Understanding JD Edwards EnterpriseOne Development Tools

This section provides overview information and discusses:

- Object Management Workbench
- Data Dictionary
- Table Design
- Business View Design
- Form Design
- Data Structure Design
- Event Rules
- System Functions
- Business Functions
- Report Design
- Workflow

Note. These topics will be discussed in more detail later in the document.

JD Edwards EnterpriseOne Development Tools

The development tools that JD Edwards EnterpriseOne Tools provides offer a powerful application development environment in which you can build and customize EnterpriseOne applications to suit your specific needs quickly and easily. By using EnterpriseOne Tools development tools you can complete a variety of tasks including:

- Design and define application objects.
- Enable applications to serve different locations and languages while sharing the same data.
- Define end-to-end processes in a user-friendly, graphical design environment.

Object Management Workbench

Object Management Workbench (OMW) manages all EnterpriseOne objects. Developers use OMW to create new objects and check out existing objects from a central development environment, copying those objects to their workstation. They can then use the development tools to change objects and check them back in for others to access.

Data Dictionary

Just as a dictionary contains work definitions, the data dictionary is a central repository that contains data item definitions and attributes. These attributes determine how a data item:

- Appears on reports and forms.
- Validates data entry within an application.
- Assigns column and row descriptions.
- Provides text for field help
- Is stored in a table.

Table Design

.

A relational database table is used to store the data that an application uses. Although a new application might use one or more tables that already exist, you can use the Table Design Aid to create new tables if the application requires it. To create a table, you select data items and then assign key fields as indexes for retrieving and updating data.

Business View Design

Business views are the link between applications and data. A business view defines the data items from one or more tables that an application uses. After you determine the data items needed by an application, you can create a new business view if you are not able to use an existing one. With business views, you can select only the data items needed in the application, which increases performance due to less data moving over the network. For example, you could create a business view that contains only employee names and addresses from a table containing all employee data.

Form Design

Use Form Design Aid (FDA) to create one or more forms for an application. A form is a graphical user interface that enables users to interact with the system. A form can be used to search and display data, as well as enter new data and modify existing data. A single application can contain one or more forms. To create an application, determine the type of form the application requires and associate each form with a business view. To design forms, you add controls such as a grid, edit fields, push buttons, and radio buttons.

Usually, a find/browse form is the first form that appears in the application. It enables the user to locate a specific record with which to work. Upon selecting a record, a subsequent form such as a fix/inspect form can be used to provide details of the record. Power forms enable you to design applications that use one single power form to locate a specific record and display its detail records on one form.

Data Structure Design

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. You use Data Structure Design to create and modify EnterpriseOne data structures.

Event Rules

Events are activities that occur on a form, such as when a user enters information into a field or exits a field by using the Tab key. Events can be initiated by the user or by the application. Event rules (ER) are logic statements that you can create and attach to events. ER is initiated when events occur at runtime. You can attach multiple event rules to one event. The various kinds of event rules include:

- Conditional statements, such as If/Else/End If.
- While loops.
- Assignments.
- Calls to business functions.
- Form or report interconnections.
- Calls to system functions.
- Table I/O operations.

System Functions

System functions are predefined sets of logic shipped with the JD Edwards EnterpriseOne product. These functions enable you to perform specialized processing without adding custom code. You use system functions within JD Edwards EnterpriseOne Report Design Aid (RDA), Form Design Aid (FDA), and Workflow. Each of these has a specific set of system functions that apply to it.

Business Functions

A business function is an encapsulated set of business rules and logic that accomplishes a specific task and can be reused by multiple applications. Business functions provide a common way to access the JD Edwards EnterpriseOne database. Master business functions provide the logic and database calls necessary to extend, edit, and commit the full transaction to the database. Third-party applications can use master business functions for full JD Edwards EnterpriseOne functionality, data validation, security, and data integrity.

You can use master business functions to update master files (such as Address Book Master and Item Master) or to update transaction files (such as sales orders and purchase orders). Generally, master file master business functions, which access tables, are simpler than transaction file master business functions, which are specific to a program. Transaction master business functions provide a common set of functions that contain all of the necessary default values and editing for a transaction file. Transaction master business functions contain logic that ensures the integrity of the transaction being inserted, updated, or deleted from the database.

APIs

APIs are routines that perform predefined tasks. JD Edwards EnterpriseOne APIs make it easier for third-party applications to interact with JD Edwards EnterpriseOne software. These APIs are functions that you can use to manipulate data types, provide common functionality, and access the database. Several categories of APIs exist, including the Common Library Routines and JD Edwards EnterpriseOne Database (JDEBASE) APIs. Programing with APIs is useful for these reasons:

- No code modifications are required as functionality is upgraded.
- When a data structure changes, source modifications are minimal to nonexistent.
- Common functionality is provided through the APIs, and they are less prone to error.
- When the code in an API changes, business functions typically only need to be recompiled and relinked.

Report Design

You can use Report Design Aid (RDA) to create a variety of simple and complex batch processes and reports. The interface is simple enough to use without programming expertise, yet powerful enough to create the most complex reports. You can also use RDA to create batch processes and reports. Report Design Aid includes a director to guide you through the process of creating report templates. This Report Director presents multiple report components from which to choose. You can create custom directors to aid in the creation of report templates. These directors are configured to use report components to meet a specific reporting requirement. After using the director to create the initial report, you can enhance the report by:

- Inserting additional report sections
- Modifying properties
- Adding logic
- Further organizing the data
- Calculating totals

The design work space in RDA can be configured to accommodate individual work preferences. You can:

- Modify the report view options.

- Select which toolbars and windows to display.
- Arrange windows.

You can use RDA with terminal server. Just like in a traditional client server configuration, a report template that is checked out using terminal server cannot be accessed by other users.

Batch Versions

Batch Versions is a tool that you use to create and process versions of report templates. You can use Batch Versions to:

- Add and copy batch versions.
- Define processing options, data selection and data sequencing.
- Check out batch versions, check in batch versions, erase the check out, and copy version specifications to the enterprise server.
- Access RDA to modify batch versions without changing the report template specifications.
- Submit batch versions for processing and override processing options, data selection, and data sequencing at runtime.
- Review batch version processing by using BrowsER, the report cover page, and logs for reporting.

Workflow

EnterpriseOne Workflow Tools enables you to automate a high-volume, formerly paper-based process into an email-based process flow across a network. Documents, information, and tasks pass from one participant to another for action based on a set of procedural rules. The result is an automated and efficient process with minimal user involvement, which enables you to streamline existing business processes, increase efficiency, and reduce process time.

CHAPTER 3

Understanding Objects

This chapter provides an overview of Oracle's JD Edwards EnterpriseOne objects and discusses how EnterpriseOne stores objects.

Understanding Objects

In JD Edwards EnterpriseOne, an object is a reusable entity that is based on software specifications created by the JD Edwards EnterpriseOne Tools.

A specification is a complete description of a JD Edwards EnterpriseOne object. Specifications can be thought of as metadata. Each object has its own specification, which is stored on both the server and the workstation. Some specifications describe types of objects; for example, data structure specifications can describe business function data structures, processing option structures, or media object structures.

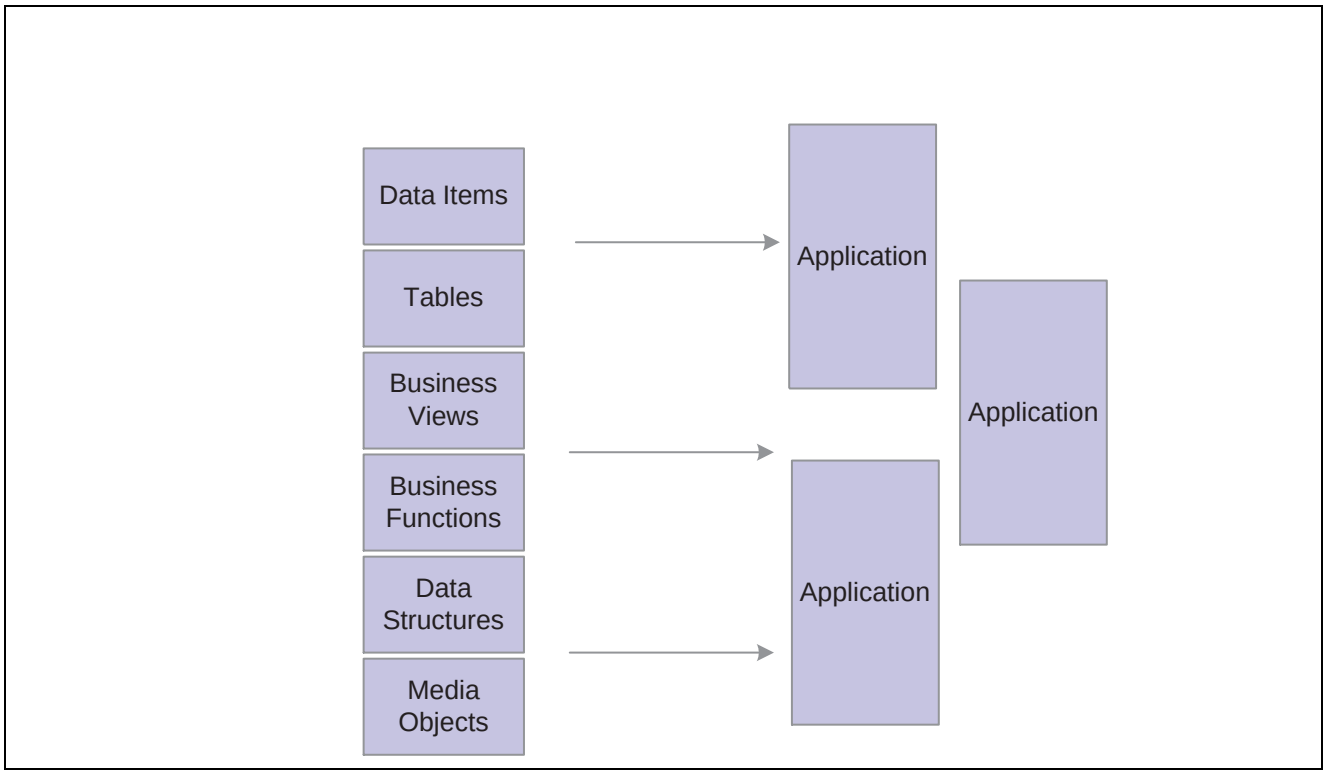
JD Edwards EnterpriseOne architecture is object-based, which means that discrete software objects are the basis for all applications and that developers can reuse the objects in multiple applications. This use of objects (applications being broken down into smaller components) allows JD Edwards EnterpriseOne to provide true distributed processing. Developers create objects using JD Edwards EnterpriseOne Tools.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Working with Objects".

Examples of JD Edwards EnterpriseOne objects include the following:

- Batch applications
- Business functions (encapsulated routines)
- Business views
- Data dictionary items
- Data structures
- Event rules
- Interactive applications
- Media objects
- Tables

This diagram shows components of the object model:



Development Objects

Understanding How JD Edwards EnterpriseOne Stores Objects

JD Edwards EnterpriseOne stores objects in the following two places:

- A central-storage server stores central objects Central objects reside in a central location from which you can deploy them. Other objects, such as specifications, are stored in a relational database. Still others, such as DLLs and source code, are stored on a file server.
- Any machine (workstation or server) that runs JD Edwards EnterpriseOne stores replicated objects A copy (replicated) set of the central objects must reside on each development workstation and server that runs JD Edwards EnterpriseOne. The path code indicates the directory in which these objects are located.

To move objects between the server and workstation, you use the check-in and check-out options in JD Edwards Object Management Workbench. When you create an object, it initially resides on your workstation. Unless you check it into the server, it is available only to you. After you check it into the server, it is available for other users to check out. When you check out an object, all object specification records (a collection of data that defines an EnterpriseOne object) are copied from the server to your workstation.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Getting Started with JD Edwards EnterpriseOne OMW".

CHAPTER 4

Understanding Creating and Managing Objects

This chapter provides an overview of Object Management Workbench.

Understanding Object Management Workbench

Object Management Workbench is the primary component of the change management system for JD Edwards EnterpriseOne development. A change management system is vital to a productive development environment because it helps organize a myriad of development activities and helps prevent problems, such as when a developer intermixes components from different releases or when multiple developers simultaneously change an object. Object Management Workbench automates many of these change management activities.

This section discusses:

- OMW projects
- Allowed actions
- Tokens
- The OMW interface
- Object Librarian and non-Object Librarian objects

OMW Projects

Projects are composed of objects and owners. All development of objects within JD Edwards EnterpriseOne must be performed within the context of a project. Usually, you must first create or select a project, add an object to it, and then you can work with that object. Typically, objects are included in a project because they have been modified or created by a developer to complete a specific task.

In addition to objects, users can be associated with different projects. In fact, before you can add an object to a project, you must have been added to the project as a user in a role that has permission to add objects. A user can be assigned to the same project more than once with different roles. Projects may also contain other projects.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Working with Projects".

Allowed Actions

Allowed actions are rules that define the actions that may be performed by a user who is assigned a specific user role. You set up these rules for each user role, object type, and project status by using the Object Management Workbench Configuration program.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Configuring User Roles and Allowed Actions".

Tokens

Some objects use tokens to minimize the possibility of one user overwriting another user's changes to an object. The token management system organizes application development by providing a single checkout environment. Tokens provide a change control solution in a system that does not support merging or multiple versions of object specifications.

Note. Only Object Librarian objects have tokens.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Working with Tokens".

The OMW Interface

From left to right, the initial OMW form displays these features:

- The project window, which displays your projects and their related objects and users. To view your current projects, click Find.
- The center column, which contains action buttons that you use to perform actions on a selected object. Available buttons vary based on your roles in the current project and on the status of the project in which the selected object resides. When you first launch OMW, no buttons appear in the center column because you have not selected an object.
- The information window, which displays a Web site; project status and release information; object or user information; and search results. Initially, the window displays a Web site or HTML page. The contents change based on your tab and object selections. For example, when you select a project or an object in the project window, the information window displays information about the selected project or object. To return this window to its initial state, click News on the toolbar.

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Understanding JD Edwards EnterpriseOne OMW," The JD Edwards EnterpriseOne OMW Interface.

Object Librarian and Non-Object Librarian Objects

OMW provides control of EnterpriseOne objects in a simple, integrated, graphical user interface for software development. In EnterpriseOne, an object is a reusable entity based on software specifications that are created by the EnterpriseOne development tools.

In OMW, this definition is expanded to include non-Object Librarian objects that are data source-based rather than path code-based.

JD Edwards EnterpriseOne objects include the following Object Librarian objects:

- Batch applications and versions
- Business functions
- Business views
- Data structures
- Interactive applications
- Media objects
- Tables

EnterpriseOne objects include the following non-Object Librarian objects:

- Data dictionary items
- User defined code items
- Workflow objects

See *JD Edwards EnterpriseOne Tools 8.97 Object Management Workbench Guide*, "Understanding JD Edwards EnterpriseOne OMW," Object Librarian and Non-Object Librarian Objects.

CHAPTER 5

Creating and Maintaining Applications

This chapter provides an overview of.

- Data Dictionary
- Table Design Aid
- Business View Design Aid
- Form Design Aid
- Report Design Aid
- Data Structure Design
- Event Rule Design
- System Functions
- Table I/O
- Business Function Design Aid
- Processing Options

Understanding Applications

An application is a collection of objects that performs a specific task. You use JD Edwards EnterpriseOne Tools to build standard groups of related applications, such as: Architecture, engineering, and construction; Distribution; Energy and chemical systems; Financial applications; Workforce management; Manufacturing; and Technical applications. These applications share a common user interface because they are all generated through JD Edwards EnterpriseOne Tools. Applications refer to both interactive and batch applications. For example, all of the following are applications:

- Address Book Revisions
- Sales Order Entry
- General Ledger Post
- Trial Balance Report

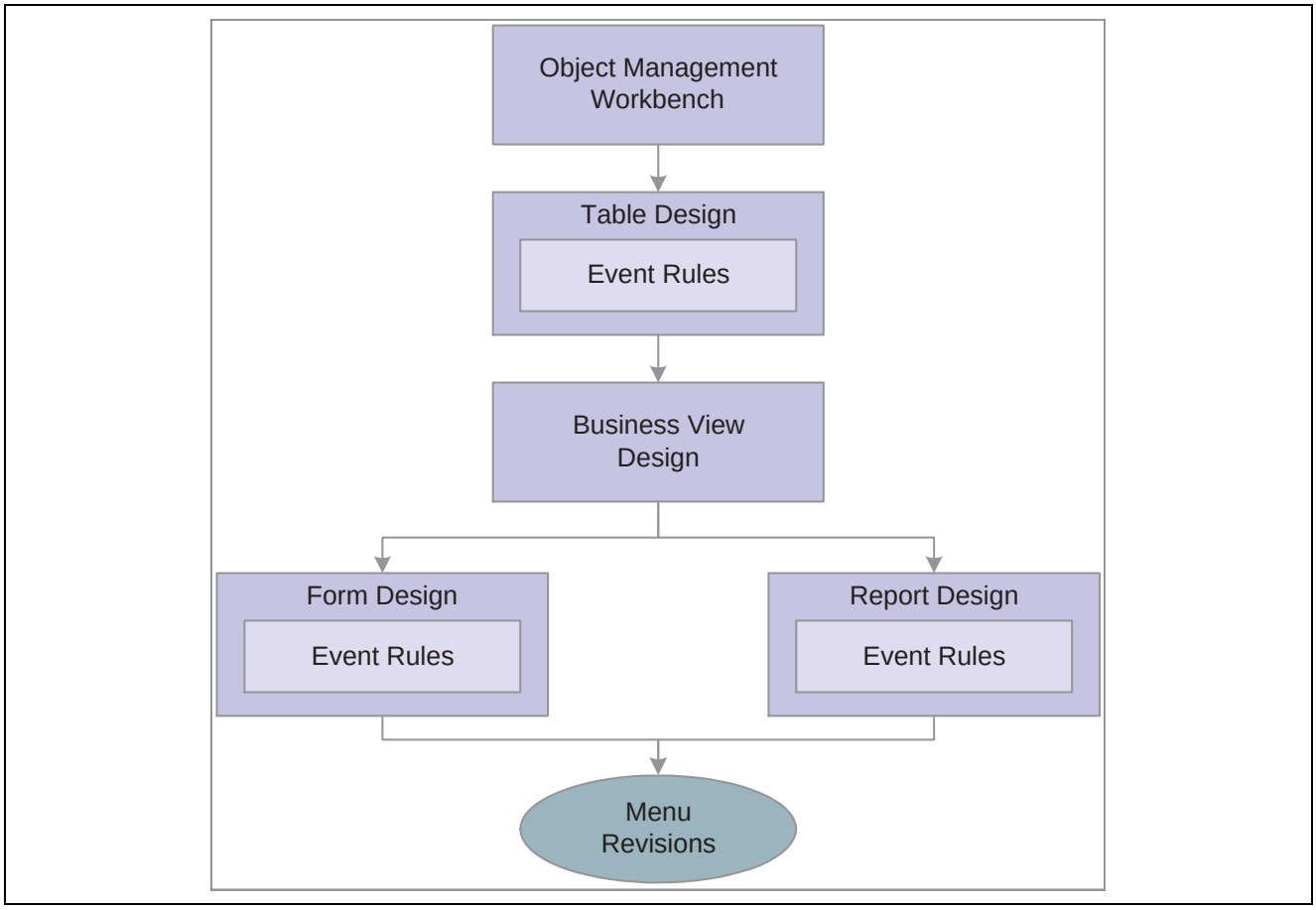
See *JD Edwards EnterpriseOne Tools 8.97 Development Guidelines for Application Design Guide*, "Getting Started with JD Edwards EnterpriseOne Tools Development Guidelines for Application Design".

Creating Applications

You use JD Edwards EnterpriseOne Tools to build your applications. You always begin your application development from the JD Edwards EnterpriseOne Object Management Workbench. You might not use every tool to create an application. For example, if you don't need to add or modify data items you would proceed to Table Design from the Object Management Workbench. If one or more existing database tables contain all of the data items that you want to include in your application, you can skip the step of designing a table and proceed to Business View Design.

This flowchart illustrates the Development Cycle:

See *JD Edwards EnterpriseOne Tools 8.97 Development Guidelines for Application Design Guide*, "Understanding Application Development Guidelines," Interactive Application Fundamentals.



Development Cycle

Understanding Data Items and the Data Dictionary

A data item identifies a unit of information. The data item definition defines how the item can be used and includes information such as the type of item and its length.

Because the data dictionary is dynamic, any changes that you make to a data item are effective immediately for all applications that include the data item. Applications access the data dictionary at runtime and immediately reflect modifications to data item attributes such as field descriptions, column headings, decimals, and edit rules.

You use the data dictionary to create, view, and update attributes for data items. You can copy a data item with similar attributes and modify it for your specific needs. This method can be quicker and easier than creating a new data item, but if you use this method you must distinguish between the original and the copy. You distinguish between them by modifying the alias.

Because changes to a data item are immediately reflected throughout the JD Edwards EnterpriseOne tools at runtime, remember that changing the type and attributes of a data item might affect how your data is stored and cause discrepancies among records.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Dictionary Guide*, "Understanding the Data Dictionary".

Understanding Table Design

A relational database table stores the data that an application uses in columns and rows. Each column is a data item, and each row is a record. You can create one or more tables for use in an application. To create a table, you select data items (the data items must already exist in the data dictionary) to include in the table and assign key fields as indices for retrieving and updating data. You must define your table so that JD Edwards EnterpriseOne software recognizes that the table exists.

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

- Create a new table.
- Add or delete a data item.
- Add or modify an index.

An index identifies records in a table. A primary index identifies unique records in a table. An index is composed of one or more keys, or data items, within the table. An index enables a database management system (DBMS) to sort and locate records quickly.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Access Tools Guide*, "Designing Tables," Understanding JD Edwards EnterpriseOne Table Design Aid.

Understanding Business View Design

A business view is a selection of data items from one or more tables. After you create a table, use Business View Design to choose only the data items that are required for your application. EnterpriseOne uses the business view that you define to generate the appropriate SQL statements necessary to retrieve data from any of the supported databases. After you define a business view, you can create a form that updates data in an interactive application or you can design a report that displays data. Because you choose only those data items that an application requires, less data moves over the network.

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

- Contain some or all of the data items from one or more tables.
- Link a JD Edwards EnterpriseOne application to one or more tables.
- Define the data items from multiple tables used by an application (such as table joins or table unions).

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Access Tools Guide*, "Understanding Business Views".

Understanding Form Design Aid

Form Design Aid is part of the Interactive Application Design and is used to create or modify EnterpriseOne applications. Applications are composed of forms, and a form is the interface between a user and a table. This interface should present the data logically and contain the functions that are necessary to enter and manipulate data.

Interactive Application Design is the entry point to several tools for creating, generating, running, maintaining, and securing applications. Interactive Application Design includes Form Design Aid for creating forms and Event Rules Design for attaching business logic through event rules. Use Interactive Application Design to do the following:

- Access Form Design for creating forms.
- Run an application
- • Create text overrides
- Browse ER
- Browse forms in an application
- Use Visual ER Compare to compare event rules between two versions of an application
- Use FDA Compare to compare one version of an application with another

To start Interactive Application Design, choose an application in Object Management Workbench and click the Design button. In Interactive Application Design, you can change the metadata for the application. To access application metadata, click the Summary, Category Codes, and Install/Merge Codes tabs. You can also attach text and files to an application by clicking the Attachments tab. You can access all other functions from the Design Tools tab.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Form Design Aid Guide*, "Getting Started with JD Edwards EnterpriseOne Tools: Form Design Aid".

Understanding Report Design Aid

Report Design Aid is used to present business data stored in the EnterpriseOne database. EnterpriseOne data is stored in databases using relational tables. The data is typically presented using batch applications that access the data through business views.

You can use Report Design Aid to create a variety of simple and complex batch processes and reports. The interface is simple enough to use without programming expertise, yet powerful enough to create the most complex reports. You can also use Report Design Aid to create batch processes and reports.

Each report is comprised of sections, which are the building blocks of all reports. Within the template, you can add, hide, remove, and rearrange sections as needed.

You cannot process a report without a batch version. The batch version is submitted for processing, and once submitted, runs without user interaction. You do not interact with the report again until processing is complete.

A report exists as a set of specifications that are read by the EnterpriseOne batch engine for processing. You can create variations of a single report template using batch versions. The first step in creating a report is to create a report object within EnterpriseOne. This report is actually a template from which multiple versions can be created.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Report Design Aid Guide*, "Getting Started with JD Edwards EnterpriseOne Report Design Aid".

Understanding Data Structure Design

Data structures are a key element of any programming language or environment. A data structure is a list of parameters that passes data among applications and tables or forms. JD Edwards EnterpriseOne uses data structures in the following instances:

- The system generates a data structure.
- You create a data structure.

The two types of system-generated data structures are as follows:

- Form

  Each form with an attached business view has a default data structure. Data structures receive parameters from or send parameters to other forms during Form Interconnects. You maintain the data structure by using the Form/Data Structure menu option in Form Design.

- Report

  A batch application with an attached business view can receive parameters from or send parameters to a data structure. You can create and maintain the data structure from the Report/Data Structure menu option in Report Design. Unlike a form data structure, this type of data structure is not automatically populated with data items.

As a user, you can create three types of data structures, as follows:

- Media object data structures

  To enable an application for media objects, you must create a data structure to pass arguments from the application table to the media object table. To work with a data structure for media objects, create a new media object data structure or select an existing one to modify in Object Management Workbench.

- Processing options data structures

  You use processing options to create an input property sheet. You use a parameter list to pass processing options to an application. You can create a processing option data structure template or modify an existing template in Object Management Workbench.

- Business function data structures

Any business function, whether it uses C or Business Function Event Rules as its source language, must have a defined data structure to send or receive parameters to or from applications. You can create a DSTR object type, or choose an existing object type to work with in Object Management Workbench. You can also create data structures for text substitution messages. Additionally, you can attach notes, such as an explanation of use, to any data structure or data item within the structure.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Structure Design Guide*, "Getting Started with JD Edwards EnterpriseOne Data Structure Design".

Understanding Event Rules Design

Use Event Rules Design to create business logic for an application. You can create event rules that do the following:

- Perform a mathematical calculation.
- Pass data from a field on a form to a field on another form.
- Count grid rows that are populated with data.
- Interconnect two forms.
- Hide or display a control using a system function.
- Evaluate If/While and Else conditions.
- Assign a value or an expression to a field.
- Create variables or programmer-defined fields at runtime.
- Perform a batch process upon completion of an interactive application.
- Process table input and output, validate data, and retrieve records.

Areas where event rules can be added are:

- Controls

  A control is a reusable object that appears on a form. Examples include push buttons, edit fields, and grids. A form itself is also considered a control. Controls can be simple or complex. Simple controls have few event points to which logic can be attached. Complex controls can have many event points to which logic can be attached

- Events

  Events are activities that occur on a form, such as entering information a form or exiting a field by using the Tab key. Events can be initiated by the user or the application. A single control might initiate multiple events. The system also initiates some events, such as Last Grid Record Read, when certain actions occur

- Form Processing

  Form processing refers to the business logic associated with each form. By default, each type of EnterpriseOne form automatically processes various events. You specify additional logic by using Event Rules Design. Form processing depends on the occurrence of specific events, such as initializing a form or changing the value of a field

- Event Rules

Event rules are logic statements that you can create and attach to events. JD Edwards EnterpriseOne software uses two types of event rules: business function event rules and embedded event rules. Event rules are initiated when events occur at runtime. You can attach multiple event rules to one event. The various kinds of event rules include:

- Business Function Event Rules

  Business function event rules are encapsulated, reusable, business logic that you create using Event Rules Design, rather than C programming. Business function event rules are stored as objects and are compiled. Business function event rules are sometimes called Named Event Rules (NERs).

- Embedded Event Rules

  Embedded event rules are specific to a particular table, interactive application, or batch application. They are not reusable. Examples include using form-to-form calls, hiding a field that is based on a value in a processing option, and calling a business function. Embedded event rules can be in application event rules (interactive or batch) or in table event rules. They can be compiled or uncompiled.

- Application Event Rules

  You can add business logic that is specific to a particular application. Interactive applications connect event rules via Form Design, while batch event rules use Report Design.

- Table Event Rules

  You can create database triggers, or rules that you attach to a table by using Table Design Event Rules. The logic that is attached to a table is run whenever any application initiates that database event. For example, to maintain referential integrity, you might attach rules to a master table that delete all children when a parent is deleted. Any application that deletes information from that table does not need to have the parent/child logic embedded in it because that logic exists in the table.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Event Rules Guide*, "Getting Started with JD Edwards EnterpriseOne Tools Development Tools: Event Rules".

Understanding System Functions

System Functions are procedures provided by the tool and are usually specific to the type of component being used. For example there are system functions to hide and show fields on an application, and there are system functions to execute different sections in a batch application. Parameters are passed into and out of the component for functionality. The available system functions are determined by the object type.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Form Design Aid Guide*, "System Functions in Form Design Aid".

Understanding Table I/O

Use the Table I/O button in Event Rules Design to create instructions that perform table input and output (I/O) so that you do not need to manually code a business function in C code. Table I/O allows you to access a table through event rules. You can use table I/O to do the following:

- Validate data
- Retrieve records

- Update or delete records across files
- Add records

For example, you can use table I/O to display information in a table that your application does not use. You can use Log Viewer to view your table I/O SQL statements in the jdedebug.log. To do so, your jde.ini file must have debugging set to File.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Access Tools Guide*, "Working with Table Input/Output".

Understanding Business Function Design

This topic discusses both C business functions and named event rules, and includes information about master business functions, Business Function Builder, and business function documentation.

You can use business functions to enhance EnterpriseOne applications by grouping related business logic. Journal Entry Transactions, Calculating Depreciation, and Sales Order Transactions are examples of business functions.

You can create business functions using one of the following methods:

- Event rules scripting language

  The business functions that you create using the event rules scripting language are referred to as Business Function Event Rules (also called Named Event Rules). If possible, use Business Function Event Rules for your business functions. In some instances, C business functions might better suit your needs.

  Note. NERs get generated into either C or Java.

- C programming code

  EnterpriseOne software creates a shell into which the user inserts logic using C. You use C business functions mainly for caching, but they can also be used for the following:

  - Batch error level messaging
  - Large functions

    C business functions work better for large functions (as determined by the group). If you have a large function, you can break the code up into smaller individual functions and call them from the larger function.
  - Functions for which performance is critical
  - Complex Select statements

After you create business functions, you can attach them to EnterpriseOne applications to provide additional power, flexibility, and control.

See *JD Edwards EnterpriseOne Tools 8.97 Development Standards for Business Function Programming Guide*, "Getting Started with JD Edwards EnterpriseOne Tools Development Standards for Business Function Programming".

Understanding Processing Options

Processing options control how an interactive or batch application processes data. You can use processing options to change the way in which an application or a report appears or behaves. You can attach unique processing options to different versions of the same application, which allows you to change the behavior of an application without creating a new application. In addition, you can use processing options to do the following:

- Control the path that a user can use to navigate through a system.
- Set up default values.
- Configure 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.

You can define processing options for an application that automatically appear at runtime. In addition, you might need to create a processing option version. The procedures for creating a processing option version are similar to those for creating an interactive version.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Structure Design Guide*, "Using Processing Options".

CHAPTER 6

Additional Topics

This chapter provides an overview of:

- Caching
- Messaging
- Transaction Processing
- Currency
- Media Objects
- Debugging
- Performance
- Cross Reference Facility

Understanding Caching

Caching is a process that stores a local copy of frequently accessed content of remote objects. Caching can improve performance. EnterpriseOne software caches information in the following two ways:

- The system automatically caches some tables, such as those associated with constants, when it reads them from the database at startup. It caches these tables to a user's workstation or to a server for faster data access and retrieval.
- Individual applications can be enabled to use cache. JDECACHE APIs allow the server or workstation memory to be used as temporary storage.

JDECACHE can hold any type of indexed data that your application needs to store in memory, regardless of the platform on which the application is running; therefore, an entire table can be read from a database and stored in memory. No limitations exist regarding the type of data, size of data, or number of data caches that an application can have, other than the limitations of the computer on which it is running. Both fixed-length and variable-length records are supported. To use JDECACHE on any supported platform, you need to know only a simple set of API calls.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: APIs and Business Functions Guide*, "Working with APIs," Working with JDECACHE.

Understanding Messaging

Use EnterpriseOne messaging features to communicate pertinent information to the end user in the most effective and user-friendly way. When you design an application to use messaging, you must evaluate what information is necessary to enable a user to accomplish a task. You can deliver a message in real time, whereupon the message is displayed in an interactive application, or you can send messages to the Employee Work Center. The method that you use to provide information to the user depends on the situation. For example, you can do the following:

- Use an interactive error message if the system encounters an error during the entry of a record.
- Use an informational message that the system sends to the Workflow Center if information needs to be conveyed and responded to.
- Use an alert message if information is urgent and requires immediate attention.
- Use a batch error message if the system detects errors in a batch process, such as while a report is running.

The three components of creating system-generated messages are as follows:

- The message itself.

  Do you require a simple message or a text substitution message? Are all of the text substitution pieces available?

- The logic that applies to the message.

  Has certain criteria, such as event rule logic, been met so that a message should be sent?

- The message type.

  Does the message require action by the users? Are all of the required parameters available at the time the message is to be sent?

Batch Error Messages

The error message system gives users a consistent interface to review errors when working with batch programs. When a batch program has finished processing all messages regarding the success or failure of a job, the system sends a message to the user in the Employee Work Center. To enhance the usability of the messages, the system uses a tree structure (or parent/child structure) to group related messages. To provide additional flexibility and functionality, you can use text substitution, and you can make a message active, meaning that the user can open an associated form by clicking within the message.

The Employee Work Center displays the error messages that appear after a batch job has completed. When you create these batch error messages, you need to determine the possible messages EnterpriseOne users will need. For example, you might create a number of different messages that are generated when a journal entry report is run. You can create a message stating that the report completed normally if the report balances. Additionally, you can create multiple levels of messages describing various errors if the report is out of balance. The first level might state that the report completed with errors, and additional levels would explain the specific details about the errors.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Batch Versions Guide*, “Getting Started with JD Edwards EnterpriseOne Batch Versions”.

Understanding Transaction Processing

A transaction is a logical unit of work (comprised of one or more SQL statements) performed on the database to complete a common task and maintain data consistency. Transaction statements are closely related and perform interdependent actions. Each statement performs part of the task, but all of them are required for the complete task.

Transaction Processing ensures that related data is added to or deleted from the database simultaneously, thus preserving data integrity in your application. In transaction processing, data is not written to the database until a commit command is issued. When this happens, data is permanently written to the database.

For example, if a transaction comprises database operations to update two database tables, either all updates will be made to both tables, or no updates will be made to either table. This condition guarantees that the data remains in a consistent state and the integrity of the data is maintained.

Using the EnterpriseOne development tools, you can enable an application for transaction processing and define which database operations comprise a transaction.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Form Design Aid Guide*, "Working with Transaction Processing".

Understanding Currency

Enterprises that do business internationally have additional accounting needs and added complexity. This complexity arises from doing business in different currencies and having to follow different reporting and accounting requirements. Some fundamental requirements for an international enterprise include:

- Conversion of foreign currencies to the local currency
- Conversion of the different local currencies into one currency for reporting and comparisons
- Adhering to regulations defined in the countries of operation
- Revaluation of currencies due to fluctuation in exchange rates

EnterpriseOne currency implementation is controlled by the developer and includes:

- Currency retrieval through database triggers and table event rules (TER).
- Business function event rules.
- System APIs for accessing cached tables.

See *JD Edwards EnterpriseOne Tools 8.97 Development Guidelines for Application Design Guide*, "Using Currency".

Understanding Media Objects

EnterpriseOne media objects and imaging features enable you to attach useful information to an application, including information that might currently exist as a paper-based document. The media objects feature enables you to attach the information to applications, forms and rows, and Object Librarian objects. The imaging feature within media objects gives you flexibility to create a more efficient method of information storage.

This table describes the types of information that you can attach to a grid row or a form:

Type	Description
Text	Media objects provide a word processor that lets you create a text-only attachment. For example, you can use a text attachment to provide specific instructions for a form or additional information about a record.
Image	Images include files such as Windows bitmaps, Graphics Interchange Format (GIF) files, and JPEG files. These files might represent electronically created files, as well as scanned images of paper-based documents.
OLE	Media objects can be files that conform to the OLE standard. OLE enables you to create links between different programs. By using these links, you can create and edit an object from one program in a different program. EnterpriseOne provides the links that you need to attach OLE objects.
Shortcuts	A shortcut is a link that opens an EnterpriseOne application. Within media objects, you can only attach EnterpriseOne shortcuts; that is, you cannot attach Windows shortcuts to media objects.
Uniform Resource Locations (URL) and files	Media objects can be links to web page URLs or other related files.

System administrators can also set up templates. A template might include attachments of its own, such as images and shortcuts. For example, you can create a letterhead and a standard form for a memo. You might create a shortcut in the template to provide access to an application that uses data specific to the information that you add to the template.

See *JD Edwards EnterpriseOne Tools 8.97 System Administration Guide*, "Setting Up Media Objects and Imaging," Understanding Media Object Processing.

Understanding Debugging

Debugging is the method you use to determine the state of your program at any point of execution. You can use debugging to help you solve problems and to test and confirm program execution.

You can use a debugger to stop program execution so you can see the state of the program at a specific point. This allows you to view the values of input parameters, output parameters, and variables at the specified point. When program execution is stopped, you can review the code line-byline to check such issues as flow of execution and data integrity.

You can use the following two tools for debugging EnterpriseOne:

- EnterpriseOne Event Rules Debugger
- Microsoft Visual C++ Debugger

You use the Event Rules Debugger to debug event rules and the following:

- Interactive applications
- Reports
- Table conversions

You use the Visual C++ Debugger to debug C business functions or NERs that are generated into C.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Access Tools Guide*, "Understanding the Cross Reference Facility".

Understanding the Cross Reference Facility

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

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

Because the cross-reference files are not automatically rebuilt when objects are created and modified, the Cross Reference Facility should be rebuilt periodically. You can regularly schedule cross-reference builds to ensure that the cross-reference information is up-to-date.

See *JD Edwards EnterpriseOne Tools 8.97 Development Tools: Data Access Tools Guide*, "Using Cross Reference Facility".

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 rational databases, flat file format, and other data or message encoding, such as XML.
alternate currency	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. 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.
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	A process of transferring records from a third-party system to JD Edwards EnterpriseOne. 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.
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	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. See also direct connect and store-and-forward.
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 EntepriseOne servers, reset statistics, and set notifications.

named event rule (NER) Encapsulated, reusable business logic created using event rules, rather that 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.

nota fiscal In Brazil, a legal document that must accompany all commercial transactions for tax purposes and that must contain information required by tax regulations.

nota fiscal factura In Brazil, a nota fiscal with invoice information.

See also *nota fiscal.*

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 The designated path for advancing objects or projects in a workflow. The following is the normal promotion cycle (path):

11>21>26>28>38>01

In this path, *11* equals new project pending review, *21* equals programming, *26* equals QA test/review, *28* equals QA test/review complete, *38* equals in production, *01* 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.

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	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. 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.
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

R

S

T

V

W