



# Oracle® Product Development Exchange Technical Reference Manual

**RELEASE 6.2**

**May 2001**

**ORACLE®**

Oracle® Product Development Exchange  
Technical Reference Manual  
Release 6.2

The part number for this book is A88861-01.

Copyright © 1994, 2001. Oracle Corporation. All rights reserved.

Major Contributors: Jeff Barton, Girish Divate, Rahul Chitko, Andrew Long, Suhail Maqsood, Joyce Fee, Varughese George, John Haddad

This Technical Reference Manual (TRM) in any form, software or printed matter, contains proprietary information of Oracle Corporation; it is provided under an Oracle Corporation agreement containing restrictions on use and disclosure and is also protected by copyright, patent, and other intellectual property law. Restrictions applicable to this TRM include, but are not limited to: (a) exercising either the same degree of care to safeguard the confidentiality of this TRM as you exercise to safeguard the confidentiality of your own most important Confidential Information or a reasonable degree of care, whichever is greater; (b) maintaining agreements with your employees and agents that protect the Confidential Information of third parties such as Oracle Corporation and instructing such employees and agents of these requirements for this TRM; (c) restricting disclosure of this TRM to those of your employees who have a "need to know" consistent with the purposes for which this TRM was disclosed to you; (d) maintaining this TRM at all times on your premises; (e) not removing or destroying any proprietary or confidential legends or markings placed upon this TRM in whatever form, software or printed matter; and (f) not reproducing or transmitting this TRM in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation. You should not use this TRM in any form, software or printed matter, to create software that performs the same or similar functions as any Oracle Corporation products.

The information in this TRM is subject to change without notice. If you find any problems in the TRM in any form, software or printed matter, please report them to us in writing. Oracle Corporation does not warrant that this TRM is error-free. This TRM is provided to customer "as-is" with no warranty of any kind. This TRM does not constitute Documentation as that term is defined in Oracle's agreements.

Reverse engineering of the Programs (software and documentation) associated with this TRM are prohibited. The Programs associated with this TRM are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be licensee's responsibility to take all appropriate fail-safe, back-up, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle disclaims liability for any damages caused by such use of the Programs.

#### Restricted Rights Legend

This TRM and the Programs associated with this TRM delivered subject to the DOD FAR Supplement are 'commercial computer software' and use, duplication and disclosure of the TRM and the Programs associated with this TRM shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, this TRM and the Programs associated with this TRM delivered subject to the Federal Acquisition Regulations are 'restricted computer software' and use, duplication and disclosure of the TRM and the Programs associated with this TRM shall be subject to the restrictions in FAR 52.227-14, Rights in Data -- General, including Alternate III (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

Oracle is a registered trademark, and CASE\*Exchange, Enabling the Information Age, Hyper\*SQL, NLS\*Workbench, Oracle7, Oracle8, Oracle 8i, Oracle Access, Oracle Application Object Library, Oracle Discoverer, Oracle Financials, Oracle Quality, Oracle Web Customers, Oracle Web Employees, Oracle Work in Process, Oracle Workflow, PL/SQL, Pro\*Ada, Pro\*C, Pro\*COBOL, Pro\*FORTRAN, Pro\*Pascal, Pro\*PL/I, SmartClient, SQL\*Connect, SQL\*Forms, SQL\*Loader, SQL\*Menu, SQL\*Net, SQL\*Plus, and SQL\*Report are trademarks or registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.



# CAUTION

**T**his Technical Reference Manual in any form -- software or printed matter -- contains proprietary, confidential information that is the exclusive property of Oracle Corporation. If you do not have a valid contract with Oracle for the use of this Technical Reference Manual or have not signed a non-disclosure agreement with Oracle covering this Technical Reference Manual, then you received this document in an unauthorized manner and are not legally entitled to possess or read it.

Use, duplication, and disclosure are subject to restrictions stated in your contract with Oracle Corporation.

L



# Contents

<b>Chapter 1</b>	<b>Introduction</b> .....	<b>1 – 1</b>
	Introduction .....	1 – 2
<b>Chapter 2</b>	<b>High-Level Design</b> .....	<b>2 – 1</b>
	Overview of High-Level Design .....	2 – 2
<b>Chapter 3</b>	<b>Detailed Design</b> .....	<b>3 – 1</b>
	Overview of Detailed Design .....	3 – 2
	Table and View Definitions .....	3 – 3





CHAPTER

# 1

## Introduction

**T**he *Oracle Product Development Exchange Technical Reference Manual* provides the information you need to understand the underlying structure of Oracle Product Development Exchange. After reading this manual, you should be able to convert your existing applications data, integrate your existing applications with your Oracle Exchange application, and write custom reports for your Oracle Exchange application, as well as read data that you need to perform other tasks.

This chapter introduces you to the *Oracle Product Development Exchange Technical Reference Manual*, and explains how to use it.

---

## Introduction

At Oracle, we design and build applications using Oracle Designer, our systems design technology that provides a complete environment to support developers through all stages of a systems life cycle. Because we use a repository-based design toolset, all the information regarding the underlying structure and processing of our applications is available to us online. Using Oracle Designer, we can present this information to you in the form of a technical reference manual.

This *Oracle Product Development Exchange Technical Reference Manual* contains detailed, up-to-date information about the underlying structure of Oracle Product Development Exchange. As we design and build new releases of Oracle Product Development Exchange applications, we update our Oracle Designer repository to reflect our enhancements. As a result, we can always provide you with an *Oracle Product Development Exchange Technical Reference Manual* that contains the latest technical information as of the publication date. Note that after the publication date we may have added new indexes to Oracle Product Development Exchange to improve performance.

### About this Manual

---

This manual describes the Oracle Applications Release 11i data model, as used by Oracle Product Development Exchange; it discusses the database we include with a fresh install of Oracle Applications Release 11i. If you have not yet upgraded to Release 11i, your database may differ from the database we document in this book.

If you have upgraded from a previous release, you might find it helpful to use this manual with the appropriate *Oracle Applications Product Update Notes* manual. The product update notes list database changes and seed data changes in Oracle Product Development Exchange between releases. The *Oracle Applications Product Update Notes Release 11* manual describes the changes between Release 10.7 and Release 11, and the *Oracle Applications Product Update Notes Release 11i* manual describes the changes between Release 11 and Release 11i.

You can contact your Oracle representative to confirm that you have the latest technical information for Oracle Product Development Exchange. You can also use *Oracle MetaLink* which is accessible through Oracle's Support Web Center ([http://www.oracle.com/support/elec\\_sup](http://www.oracle.com/support/elec_sup)).



---

## Finding the Latest Information

The *Oracle Product Development Exchange Technical Reference Manual* contains the latest information as of the publication date. For the latest information we encourage you to use *OracleMetaLink* which is accessible through Oracle's Support Web Center ([http://www.oracle.com/support/elec\\_sup](http://www.oracle.com/support/elec_sup)).

---

## Audience

The *Oracle Product Development Exchange Technical Reference Manual* provides useful guidance and assistance to:

- Technical End Users
- Consultants
- Systems Analysts
- System Administrators
- Other MIS professionals

This manual assumes that you have a basic understanding of structured analysis and design, and of relational databases. It also assumes that you are familiar with Oracle Application Object Library and your Oracle Exchange application. If you are not familiar with the above products, we suggest that you attend one or more of the training classes available through Oracle Education (see: Other Information Sources: page 1 – 8).

---

## How This Manual is Organized

This manual contains two major sections, High-Level Design and Detailed Design.

### High-Level Design

---

This section, Chapter 2, contains database diagrams and lists each database table and view that Oracle Product Development Exchange uses. This chapter also has a list of modules.



## **Detailed Design**

---

This section, Chapter 3, contains a detailed description of the Oracle Product Development Exchange database design, including information about each database table and view you might need for your custom reporting or other data requirements.



---

## How to Use This Manual

The *Oracle Product Development Exchange Technical Reference Manual* is a single, centralized source for all the information you need to know about the underlying structure and processing of your Oracle Exchange application. For example, you can use this manual when you need to:

- Convert existing application data
- Integrate your Oracle Product Development Exchange application with your other applications systems
- Write custom reports
- Define alerts against Oracle Applications tables
- Configure your Oracle Self-Service Web Applications
- Create views for decision support queries using query tools
- Create business views for Oracle Discoverer

You need not read this manual cover to cover. Use the table of contents and index to quickly locate the information you need.

---

## How Not To Use This Manual

### **Do not use this manual to plan modifications**

You should not use this manual to plan modifications to your Oracle Exchange application. Modifying Oracle Product Development Exchange limits your ability to upgrade to future releases of your Oracle Product Development Exchange application. In addition, it interferes with our ability to give you the high-quality support you deserve.

We have constructed your Oracle Exchange application so that you can customize it to fit your needs without programming, and you can integrate it with your existing applications through interface tables. However, should you require program modifications, you should contact our support team (see: Other Information Sources: page 1 – 8). They can put you in touch with Oracle Services, the professional consulting organization of Oracle. Their team of experienced applications professionals can make the modifications you need while ensuring upward compatibility with future product releases.



### **Do not write data into non-interface tables**

---

Oracle reserves the right to change the structure of Oracle Applications tables, and to change the meaning of, add, or delete lookup codes and data in future releases. Do not write data directly into or change data in non-interface tables using SQL\*Plus or other programming tools because you risk corrupting your database and interfering with our ability to support you.

Moreover, this version of the *Oracle Product Development Exchange Technical Reference Manual* does not contain complete information about the dependencies between Oracle Product Development Exchange applications tables. Therefore, you should write data into only those tables we identify as interface tables. If you write data into other non-interface tables, you risk violating your data integrity since you might not fulfill all the data dependencies in your Oracle Product Development Exchange application.

You are responsible for the support and upgrade of the logic within the procedures that you write, which may be affected by changes between releases of Oracle Applications.

### **Do not rely on upward compatibility of the data model**

---

Oracle reserves the right to change the structure of Oracle Product Development Exchange applications tables, and to change the meaning of, add, or delete lookup codes and other data in future releases. We do not guarantee the upward compatibility of the Oracle Product Development Exchange applications data model. For example, if you write a report that identifies concurrent requests that end in Error status by selecting directly from Oracle Application Object Library tables, we do not guarantee that your report will work properly after an upgrade.

---

## **About Oracle Application Object Library**

The *Oracle Product Development Exchange Technical Reference Manual* may contain references to tables that belong to Oracle Application Object Library. Oracle Application Object Library is a collection of pre-built application components and facilities for building Oracle Applications and extensions to Oracle Applications. Oracle Application Coding Standards use the Oracle Application Object Library and contains shared components including but not limited to -- forms, subroutines, concurrent programs and reports, database tables and objects, messages, menus, responsibilities, flexfield definitions and online help.

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**



**Attention:** Oracle does not support *any* customization of Oracle Application Object Library tables or modules, not even by Oracle consultants. (Oracle Application Object Library tables generally have names beginning with FND\_%.)

Accordingly, this manual does not contain detailed information about most Oracle Application Object Library tables used by your Oracle Exchange application.

---

## A Few Words About Terminology

The following list provides you with definitions for terms that we use throughout this manual:

### **Relationship**

---

A relationship describes any significant way in which two tables may be associated. For example, rows in the Journal Headers table may have a one-to-many relationship with rows in the Journal Lines table.

### **Database Diagram**

---

A database diagram is a graphic representation of application tables and the relationships between them.

### **Summary Database Diagram**

---

A summary database diagram shows the most important application tables and the relationships between them. It omits tables and relationships that contribute little to the understanding of the application data model. Typically, a summary database diagram shows tables that contain key reference and transaction data.

### **Module**

---

A module is a program or procedure that implements one or more business functions, or parts of a business function, within an application. Modules include forms, concurrent programs and reports, and subroutines.

### **Application Building Block**

---

An application building block is a set of tables and modules (forms, reports, and concurrent programs) that implement closely-related

database objects and their associated processing. Said another way, an application building block is a logical unit of an application.

---

### **QuickCodes**

QuickCodes let you define general purpose, static lists of values for window fields. QuickCodes allow you to base your program logic on lookup codes while displaying user-friendly names in a list of values window. QuickCodes simplify name and language changes by letting you change the names your end users see, while the codes in your underlying programs remain the same.

---

### **Form**

A form is a module comprised of closely related windows that are used together to perform a task. For example, the Enter Journals form in Oracle General Ledger includes the Enter Journals window, the Batch window, and the More Actions window among others. The Enter Journals window is the main window, and from it, you can use buttons to navigate to other windows in the form. The form name usually corresponds to the main window in the form, and is frequently a window you open directly from the Navigator.

---

## **Other Information Sources**

There are additional information sources, including other documentation, training and support services, that you can use to increase your knowledge and understanding of Oracle Designer, Oracle Application Object Library, and your Oracle Exchange application. We want to make these products easy for you and your staff to understand and use.

---

### **Oracle Designer Online Documentation**

The online help for Oracle Designer describes how you can use Oracle Designer for your development needs.

---

### **Oracle Applications Developer's Guide**

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User*

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**

*Interface Standards.* It also provides information to help you build your custom Developer forms so that they integrate with Oracle Applications.

---

### **Oracle Applications User Interface Standards**

This manual contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built using Oracle Forms 6.

---

### **Oracle Product Development Exchange applications User Guide**

Your user guide provides you with all the information you need to use your Release 11i Oracle Product Development Exchange application. Each user guide is organized for fast, easy access to detailed information in a function- and task-oriented organization.

---

### **Oracle Applications Flexfields Guide**

This guide provides flexfields planning, setup and reference information for the Oracle Product Development Exchange implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This manual also provides information on creating custom reports on flexfields data.

---

### **Oracle Workflow Guide**

This manual explains how to define new workflow business processes as well as customize existing Oracle Applications-embedded workflow processes. You also use this guide to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes.

---

### **Oracle Alert User Guide**

This manual explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

---

### **Multiple Reporting Currencies in Oracle Applications**

If you use the Multiple Reporting Currencies feature to report and maintain accounting records in more than one currency, use this manual before implementing Oracle Product Development Exchange.



This manual details additional steps and setup considerations for implementing Oracle Product Development Exchange with this feature.

---

### **Multiple Organizations in Oracle Applications**

If you use the Oracle Applications Multiple Organization Support feature to use multiple sets of books for one Oracle Product Development Exchange installation, use this guide to learn about setting up and using Oracle Product Development Exchange with this feature. This book describes the Oracle Applications organization model, which defines business units and the relationships between them in an arbitrarily complex enterprise. Functional and technical overviews of multiple organizations are presented, as well as information about how to set up and implement this feature set in the relevant Oracle Applications products.

---

### **Oracle Manufacturing, Distribution, Sales and Service Open Interfaces Manual**

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes open interfaces found in Oracle Manufacturing.

---

### **Oracle Applications Messages Manual**

The Oracle Applications Messages Manual contains the text of numbered error messages in Oracle Applications. (Oracle Applications messages begin with the prefix "APP-".) It also provides information on the actions you take if you get a message. Note: This manual is available only in HTML format.

## **Installation and System Administration**

---

### **Oracle Applications Installation Release Notes**

This manual contains a road map to the components of the release, including instructions about where to access the Release 11i documentation set.

---

### **Oracle Applications Concepts**

Designed to be the first book the user reads to prepare for an installation of Oracle Applications. It explains the technology stack,

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**



architecture, features and terminology for Oracle Applications Release 11i. This book also introduces the concepts behind and major uses of Applications-wide features such as MRC, BIS, languages and character sets (NLS, MLS), BIS, Self-Service Web Applications and so on.

---

### **Installing Oracle Applications**

Describes the One-Hour Install process, the method by which Release 11i will be installed. This manual includes all how-to steps, screen shots and information about Applications-wide post-install tasks.

---

### **Using the AD Utilities**

This manual contains how-to steps, screen shots and other information required to run the various AD utilities such as AutoInstall, AutoPatch, AD Administration, AD Controller, Relink and so on. It also contains information about when and why you should use these utilities.

---

### **Upgrading Oracle Applications**

This manual contains all the product specific pre- and post-upgrade steps that are required to upgrade products from Release 10.7 (NCA, SC and character-mode) or Release 11 of Oracle Applications. This manual also contains an overview chapter that describes all the tasks necessary to prepare and complete a upgrade of Oracle Applications.

---

### **Oracle Applications System Administrator's Guide**

This manual provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and manage concurrent processing.

---

### **Oracle Applications Product Update Notes**

This book contains a summary of each new feature we added since Release 11, as well as information about database changes and seed data changes that may affect your operations or any custom reports you have written. If you are upgrading from Release 10.7 you also need to read *Oracle Applications Product Update Notes Release 11*.

---

### **Oracle Self-Service Web Applications Implementation Manual**

This manual describes the setup steps for Oracle Self-Service Web Applications and the Web Applications Dictionary.



## **Oracle Applications Implementation Wizard User Guide**

---

If you are implementing more than one Oracle product, you can use the Oracle Applications Implementation Wizard to coordinate your setup activities. This guide describes how to use the wizard.

## **Other Information**

### **Training**

---

Oracle Education offers a complete set of training courses to help you and your staff master Oracle Applications. We can help you develop a training plan that provides thorough training for both your project team and your end users. We will work with you to organize courses appropriate to your job or area of responsibility.

Training professionals can show you how to plan your training throughout the implementation process so that the right amount of information is delivered to key people when they need it the most. You can attend courses at any one of our many Educational Centers, or you can arrange for our trainers to teach at your facility. In addition, we can tailor standard courses or develop custom courses to meet your needs.

### **Support**

---

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep your Oracle Exchange application working for you. This team includes your Technical Representative, Account Manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle server, and your hardware and software environment.

---

## **About Oracle**

Oracle Corporation develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 75 software modules for financial management, supply chain management, manufacturing, project systems, human resources, and sales and service management.

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**

Oracle products are available for mainframes, minicomputers, personal computers, network computers, and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.

---

## Thank You

Thanks for using Oracle Product Development Exchange applications and this technical reference manual!

We appreciate your comments and feedback. At the back of this manual is a Reader's Comment Form that you can use to explain what you like or dislike about your Oracle Exchange application or this technical reference manual. Mail your comments to the following address or call us directly at (650) 506-7000.

Oracle Applications Documentation Manager  
Oracle Corporation  
500 Oracle Parkway  
Redwood Shores, California 94065  
U.S.A.





CHAPTER

# 2

## High-Level Design

**T**his chapter presents a high-level design for Oracle Product Development Exchange that satisfies the business needs we specify during Strategy and Analysis. It contains database diagrams for Oracle Product Development Exchange application building blocks, lists of database tables and views, and a list of modules.



---

## Overview of High-Level Design

During High-Level Design, we define the application components (tables, views, and modules) we need to build our application. We specify what application components should do without specifying the details of *how* they should do it.

You can refer to this High-Level Design chapter to quickly acquaint yourself with the tables, views, and modules that comprise Oracle Product Development Exchange applications. And, you can prepare yourself to understand the detailed design and implementation of Oracle Product Development Exchange.

---

## Database Diagrams

The Database Diagrams section graphically represents all Oracle Product Development Exchange applications tables and the relationships between them, organized by building block.

Use this section to quickly learn what tables each Oracle Product Development Exchange application building block uses, and how those tables interrelate. Then, you can refer to the Table and View Definitions sections of Chapter 3 for more detailed information about each of those tables.

---

## Table Lists

The Table List sections list the Oracle Product Development Exchange applications tables. Because a product might not include at least one table for each type, this Technical Reference Manual might not include each of the following sections.

### Public Tables

---

Use the Public Table List section to quickly identify the tables you are most interested in. Then, you can refer to the Table and View Definitions sections of Chapter 3 for more detailed information about those tables.

In addition, this manual may contain full documentation for one or more of the following Application Object Library tables: FND\_DUAL, FND\_CURRENCIES, and FND\_COMMON\_LOOKUPS.



---

### **Internal Tables**

This section includes a list of private, internal tables used by Oracle Product Development Exchange; we do not provide additional documentation for these tables.

---

## **View Lists**

The View List sections list the Oracle Product Development Exchange views, with one section for each type of view. Because a product might not include at least one view for each type, this Technical Reference Manual might not include each of the following sections.

Use this section to quickly identify the views you are most interested in. Then, you can refer to the Table and View Definitions sections of Chapter 3 for more detailed information about those views.

---

### **Public Views**

This section lists views that may be useful for your custom reporting or other data requirements. The list includes a description of the view, and the page in Chapter 3 that gives detailed information about the public view.

---

### **Web Views**

This section lists views that you may need to configure your Self-Service Web applications. The list includes a description of the view, and the page in Chapter 3 that gives detailed information about the web view.

---

### **Internal Views**

This section includes each private, internal view that Oracle Product Development Exchange uses.

---

### **Multiple Reporting Currency Views**

This list includes views that were created to support the Multiple Reporting Currencies feature.

---

## Module List

The Module List section briefly describes each of the Oracle Product Development Exchange applications modules. This section lists forms, reports, and concurrent programs.

A form is a module comprised of closely related windows that are used together to perform a task. For example, the Enter Journals form in Oracle General Ledger includes the Enter Journals window, the Batch window, and the More Actions window. The Enter Journals window is the main window, and from it, you can use buttons to navigate to other windows in the form. The form name usually corresponds to the main window in the form, and is frequently a window you can open directly from the Navigator.

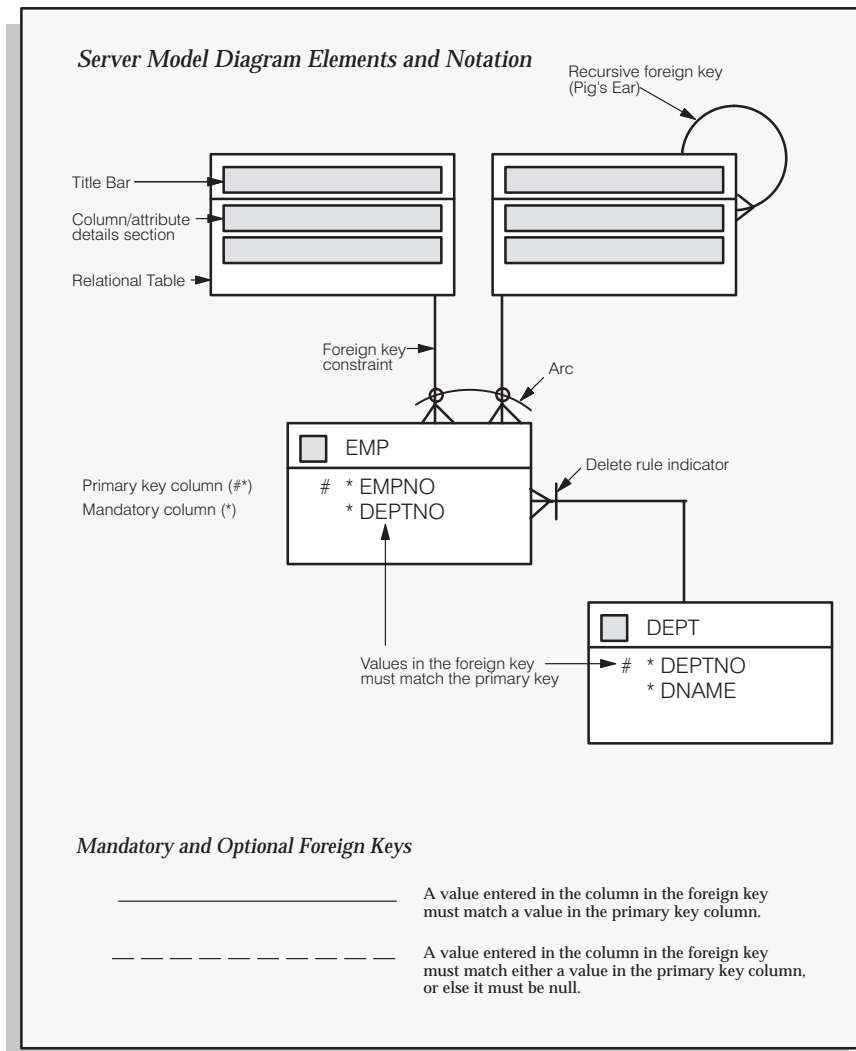
The Reports and Concurrent Programs lists include processes you can submit from the Submit Requests window or other windows, as well as processes that are submitted automatically by Oracle Product Development Exchange. Use your user guide to learn more about reports and concurrent processes.



## Database Diagramming Conventions

We use the following notational conventions in our database diagrams:

Figure 2 - 1  
Database Diagram  
Conventions





**Tables** – are the basic unit of storage in the database. A hand symbol preceding the title in the table's title bar indicates that the table is not owned by this application but shared with another.

**Foreign key constraint** – is a type of referential integrity constraint for checking the integrity of data entered in a specific column or set of columns. This specified column or set of columns is known as the foreign key.

**Delete rule indicator** – determines the action to be taken when an attempt is made to delete a related row in a join table. A line through the foreign key constraint, as shown on the above diagram, indicates that this action is restricted.

**Arcs** – specify that, for any given row in a table, a value must be entered in one of the arc columns. The remaining columns within the arc must be null.



---

## Database Diagrams

This section graphically represents most of the significant Oracle Product Development Exchange applications tables and the relationships between them, organized by building block. Use this section to quickly learn what tables each Oracle Product Development Exchange application building block uses, and how these tables interrelate. Then, you can refer to the Table and View Definitions sections of Chapter 3 for more detailed information about each of those tables.

This section contains a database diagram for each of the following Oracle Product Development Exchange application building blocks:

- Diagram 1: PDX Product Structures
- Diagram 2: PDX Change Management
- Diagram 3: PDX Item Information

Some tables, especially important reference tables, appear in more than one database diagram. When several building blocks use a table, we show that table in each appropriate database diagram.

### Product Structures

---

Diagram 2 shows the tables and relationships associated with OPDX product structure header information, product structure components, product structure revisions and types.

### Change Management

---

Diagram 3 shows the tables and relationships in change management.

### Items

---


Diagram 4 shows the tables and relationships related to OPDX items.

---

## How to Use These Database Diagrams

Here is an example of how you might use these database diagrams:

Suppose you wish to see how a change order is created for a project and its task. You would go to the Change Management diagram and find that a change order (IPD\_CHANGE\_HEADERS table) can have multiple change order lines (IPD\_CHANGE\_LINES), such as one for

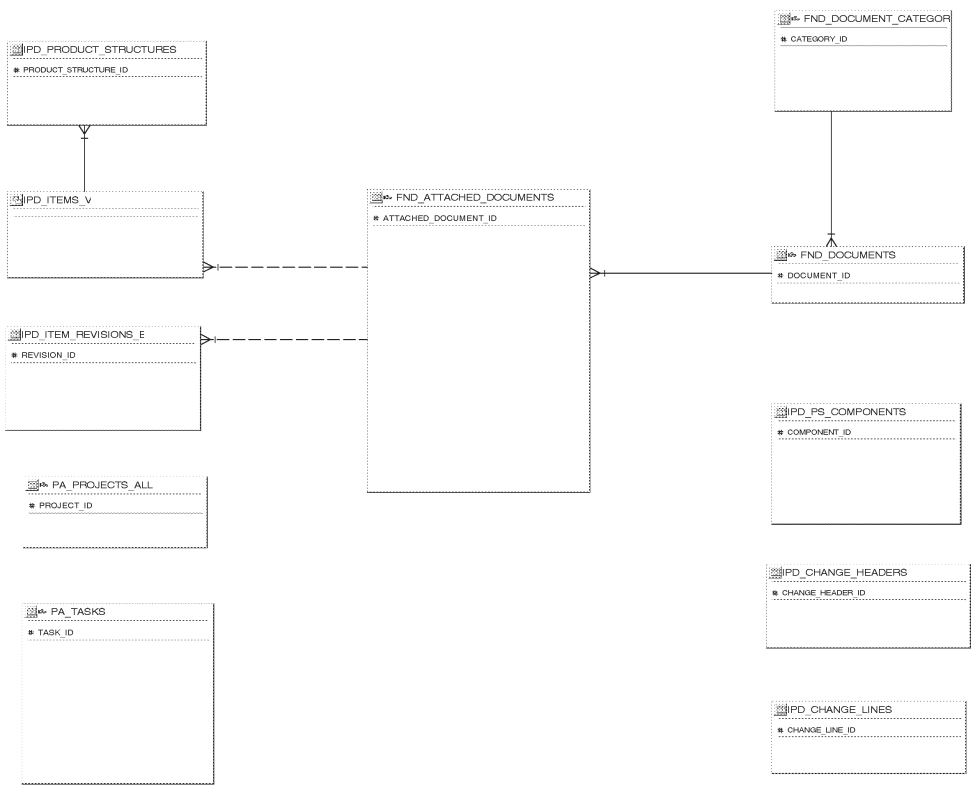


the project change and another for the task change. To see the different approval methods for the task and project changes, you would find a diagram that describes the workflows submitted (IPD\_CHANGE\_SUBMIT\_REVISIONS) for each line (IPD\_CHANGE\_LINES).



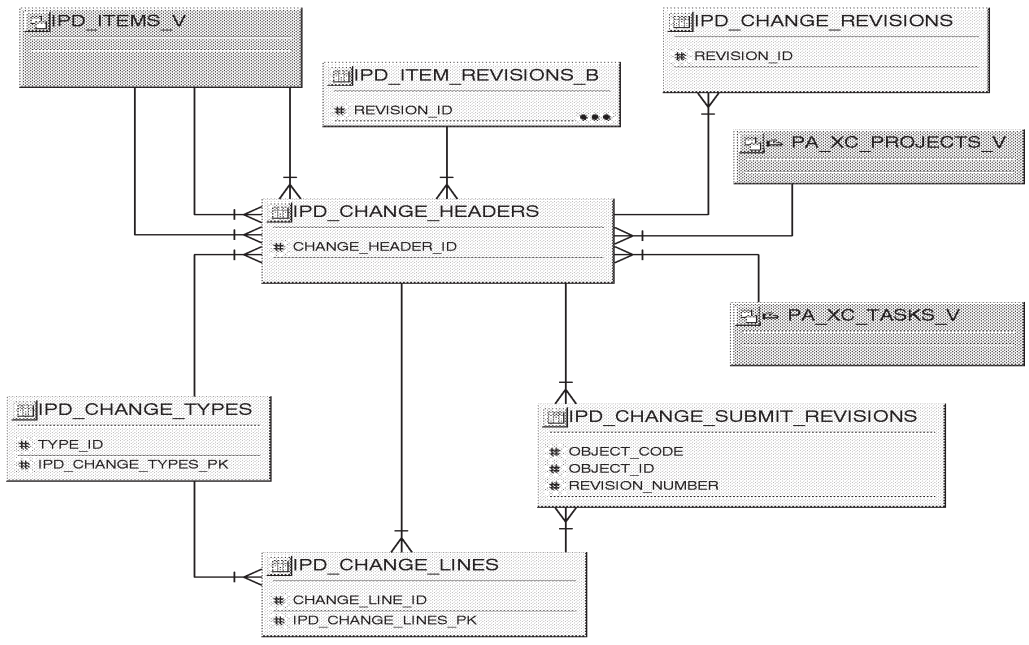
---

## PDX Product Structures



---

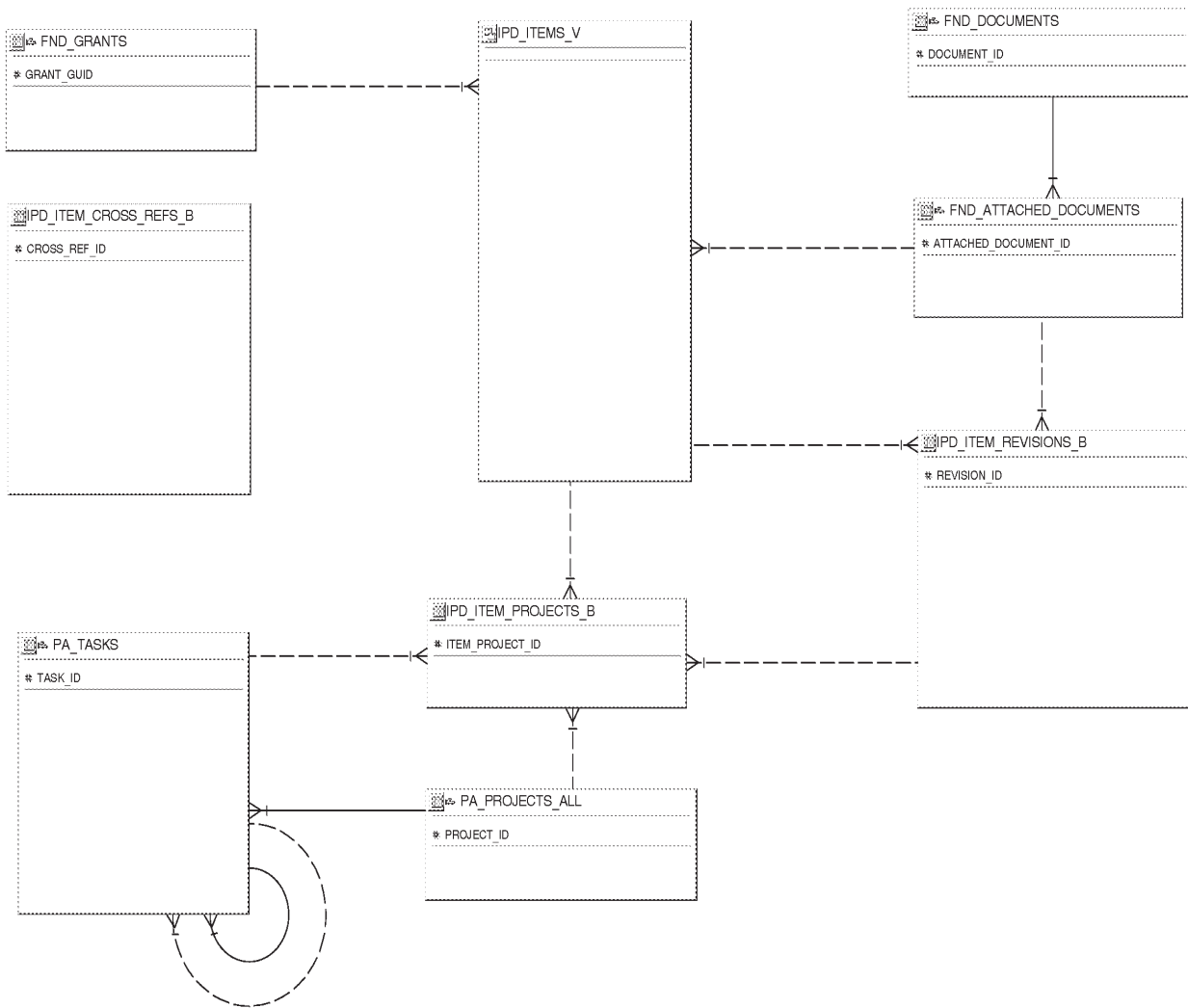
## PDX Change Management





---

## PDX Item Information



---

## Public Table List

This section lists each public database table that Internal Product Development uses and provides a brief description of each of those tables. The page reference is to the table description in Chapter 3.

Note that "public" tables are not necessarily intended for write access by custom code; Oracle Corporation supports write access using only standard Oracle Applications forms, reports, and programs, or any SQL write access to tables explicitly documented as API tables. For more information, see the "How Not To Use This Manual" section of this book's Introduction.

Internal Product Development uses the following Public tables:

<b>Table Name</b>	<b>Description</b>
ENI_MONTHS	Stores the last day of the calendar months from 1951 to 2010
FND_ATTACHED_DOCUMENTS	Attached documents
FND_DOCUMENTS	Documents
FND_DOCUMENT_CATEGORIES	Document categories
FND_GRANTS	Stores user's role grants on objects
FND_MENUS	New menu table for Release 10SC
FND_OBJECTS	Stores all object information
FND_OBJECT_INSTANCE_SETS	Stores object instance set information
ICX_POR_ITEMS	Stores information and root descriptors for non translatable data for each item in the catalog. Root descriptor columns are in the format Axxx where xxx is an integer. Some of the root descriptors are seeded by Oracle while others may be generated through the use of the application.
IPD_CHANGE_HEADERS	Change Headers Table
IPD_CHANGE_HEADERS_TL	Change Headers Table by language
IPD_CHANGE_LINES	Change Lines Table
IPD_CHANGE_LINES_TL	Change Lines Table by language
IPD_CHANGE_REVISIONS	Change Revisions Table
IPD_CHANGE_REVISIONS_TL	Change Revisions Table

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**



IPD_CHANGE_SUBMIT_REVISIONS	Change Workflow submissions History Table
IPD_CHANGE_TYPES	Change Types Table
IPD_CHANGE_TYPES_TL	Change Types Table by language
IPD_CONFERENCE_HEADERS	Conference Header Table
IPD_EVENTS	IPD_EVENTS stores event information
IPD_EVENT_NOTIFICATIONS	IPD_EVENTS stores event information
IPD_EVENT_SUBSCRIPTIONS	IPD_EVENT_SUBSCRIPTIONS stores subscription information
IPD_EVENT_TYPES_B	Stores information about event types
IPD_EVENT_TYPES_TL	Stores event types translation information
IPD_FND_OBJECTS_EXT	IPD_FND_OBJECTS_EXT stores objects associated with view privilege
IPD_ITEM_CROSS_REFS_B	Item Cross References
IPD_ITEM_CROSS_REFS_TL	Translations for IPD_ITEM_CROSS_REFS_B
IPD_ITEM_PROJECTS_B	Item / Revision to Project / Task Associations
IPD_ITEM_PROJECTS_TL	Translations for IPD_ITEM_PROJECTS_B
IPD_ITEM_REVISIONS_B	Item Revisions
IPD_ITEM_REVISIONS_TL	Translations for IPD_ITEM_REVISIONS_B
IPD_OBJECT_ROLES	IPD_OBJECT_ROLES
IPD_PRODUCT_DEFECTS	Product Defects Table
IPD_PRODUCT_STRUCTURES	Product Structure
IPD_PRODUCT_STRUCTURES_TL	Translations for Product Structures
IPD_PS_COMPONENTS	Product Structure Components
IPD_PS_COMPONENTS_INTERFACE	Product Structure Components interface table
IPD_PS_EXPLOSIONS	Product Structure Explosions
IPD_PS_HEADER_INTERFACE	Product Structure interface table
IPD_PS_PROCESS_ASSIGNMENTS	Product Structure Process Assignments

IPD_PS_REVISIONS	Product Structure Revisions
IPD_PS_REVISIONS_INTERFACE	Product Structure Revision Interface table
IPD_PS_REVISIONS_TL	Translations for IPD_PS_REVISIONS
IPD_PS_TYPES	Product Structure Types
IPD_PS_TYPES_TL	Translations for IPD_PS_TYPES
PA_ITEM_PROJECT_ASSOCIATION_V	View of Items linked with Project or Task
PA_PROJECTS_ALL	Information about projects
PA_TASKS	User-defined subdivisions of project work
PA_XC_PROJECTS_V	View List of Projects for a Specified Login User
PA_XC_PROJECT_DETAILS_V	View of all Project Details
PA_XC_PROJECT_PEOPLE_V	View List of People for a Specified Project
PA_XC_TASKS_ALL_V	View All Task Details



---

## Public View List

This section lists each public database view that Oracle Product Development Exchange uses and provides a brief description of each of those views. These views may be useful for your custom reporting or other data requirements. The page reference is to the detailed view description in Chapter 3.

Oracle Product Development Exchange uses the following public views:

<b>View Name</b>	<b>Description</b>
ENI_CO_HEADER_V	Intelligence Change Order Headers view (See page 3 – 8)
ENI_PRODUCT_STRUCTURES_V	Intelligence Product Structures view (See page 3 – 11)
ENI_PS_COMPLEXITY_V	Intelligence Product Structures Complexity view (See page 3 – 13)
ENI_PS_REUSE_V	Intelligence Product Structures Reuse view (See page 3 – 14)
IPD_CONFERENCE_DETAILS_VL	Conference Details view
IPD_CONFERENCE_HEADERS_V	Conference Headers view
IPD_FND_DOCUMENTS_V	IPD_FND_DOCUMENTS_V is view on IPD_FND_DOCUMENTS_V, provides aliases (See page 3 – 35)
IPD_FND_OBJECTS_V	IPD_FND_OBJECTS_V is view of all fnd objects. (See page 3 – 38)
IPD_ITEMS_V	PDX Items view (See page 3 – 39)
IPD_ITEM_CROSS_REFS_ALL_V	Items Cross References view (See page 3 – 40)
IPD_ITEM_CROSS_REFS_VL	Items Cross References view by language (See page 3 – 45)
IPD_ITEM_PROJECTS_VL	Item/Revision – Project/Task Association view (See page 3 – 48)
IPD_ITEM_REVISIONS_VL	Item/Revision view by language (See page 3 – 52)
IPD_PRODUCT_DEFECTS_V	Product defects view (See page 3 – 56)

---

## Forms and Table View List

This section lists supplementary views that are not essential to the Release 11i data model, but simplify coding or improve performance of Developer. For example, many of these views are used as base tables in Oracle Product Development Exchange forms.



**Warning:** We do not recommend you query or alter data using these views. Furthermore, these views may change dramatically in subsequent minor or major releases of Oracle Product Development Exchange.

Oracle Product Development Exchange uses the following Forms and Table views:

- IPD\_CHANGE\_HEADERS\_V
- IPD\_CHANGE\_HEADERS\_VL
- IPD\_CHANGE\_LINES\_V
- IPD\_CHANGE\_LINES\_VL
- IPD\_CHANGE\_REVISIONS\_VL
- IPD\_CHANGE\_TYPES\_V
- IPD\_CHANGE\_TYPES\_VL
- IPD\_GROUPS\_V
- IPD\_GROUP\_MEMBERS\_V
- IPD\_PEOPLE\_V
- IPD\_PRODUCT\_STRUCTURES\_VL
- IPD\_PS\_REVISIONS\_VL
- IPD\_PS\_TYPES\_VL







CHAPTER

# 3

## Detailed Design

**T**his chapter presents a detailed design for implementing Oracle Product Development Exchange. It contains detailed definitions of tables and views that you may need to reference to write custom reports or use for other data extraction.



---

## Overview of Detailed Design

During Detailed Design, we specify in detail how each applications component should work. We prepare detailed definitions of tables and views.

You can refer to this Detailed Design chapter to gain a detailed understanding of the underlying structure and processing of Oracle Product Development Exchange that enables you to:

- Convert existing application data
- Integrate your Oracle Product Development Exchange application with your other applications systems
- Write custom reports
- Define alerts against Oracle Applications tables
- Create views for decision support queries using query tools
- Configure your Oracle Self-Service Web Applications

---

## Table and View Definitions

The Table and View Definitions section contains a detailed definition of Oracle Product Development Exchange applications tables. For each table, it provides information about primary keys, foreign keys, QuickCodes, indexes, triggers, and sequences. It also gives you a detailed description of each column and its characteristics. In addition, it provides the SQL statement that defines each view. Review this section to get a detailed understanding of what tables your Oracle Product Development Exchange application contains, and how it uses them to hold and access the information it needs.



---

## Table and View Definitions

This section contains a detailed description of each Oracle Product Development Exchange table and view that you may need to reference. For each table, it presents detailed information about:

- Primary keys
- Foreign keys
- Column descriptions
- Indexes
- Oracle sequences
- Triggers
- View derivations

The following sections appear in each table or view description:

---

### Foreign Keys

To help you understand the relationships between tables, we list each foreign key contained in a table. For each foreign key in a table, we list the primary key table name (the table to which a foreign key refers), its corresponding primary key columns, and the foreign key columns that refer to those primary key columns.

When the primary key table has a composite primary key, we list each column of the composite key sequentially.

If a table contains two or more distinct foreign keys that refer to the same primary key table, we repeat the primary key table name and list each of the distinct foreign keys separately.

---

### QuickCodes Columns

When a database column contains a QuickCodes value, which we implement using a foreign key to FND\_LOOKUPS, MFG\_LOOKUPS, or to some other lookup table, we list the QuickCodes type (lookup type) to which the QuickCodes value must belong and a complete list of QuickCodes values and meanings. Some QuickCodes can be defined by you in the application. These values are designated as User-defined.

---

## Column Descriptions

We list the important characteristics of each column in a table or view. These characteristics include whether the column is part of the table's primary key, whether Oracle8i requires a value for this column, and the data type of the column. We also give you a brief description of how your Oracle Exchange application uses the column.

When a column is part of a table's primary key, we append the notation (PK) to the name of that column.

To help you understand which columns your Oracle Exchange application uses and which columns it does not use, we alert you to any unused column. When no module uses a database column, we show one of the following legends in the Description column:

<b>Not currently used</b>	your Oracle Exchange application does not use this column, although the column might be used in a future release.
<b>No longer used</b>	your Oracle Exchange application no longer uses this column. AutoInstall installs this column. Subsequent versions of your Oracle Exchange application might not include this column.
<b>No longer installed</b>	your Oracle Exchange application no longer uses this column. If you <i>upgraded</i> your software from an earlier version, you may still have this column, depending upon whether you chose to delete it during an upgrade process. If you <i>install</i> your Oracle Exchange application, you do not have this column.

### Standard Who Columns

---

Most Oracle Product Development Exchange applications tables contain standard columns to support \ **Row Who**. When your program or SQL\*Plus command selects a row from a table, use these columns to determine who last updated the row. If your program or SQL\*Plus command updates or inserts a row in an interface table, you must populate each of the five standard Who columns:

LAST_UPDATE_DATE	Date when a user last updated this row
LAST_UPDATED_BY	User who last updated this row (foreign key to FND_USER.USER_ID)
CREATION_DATE	Date when this row was created

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**



<b>CREATED_BY</b>	User who created this row (foreign key to FND_USER.USER_ID)
<b>LAST_UPDATE_LOGIN</b>	Operating system login of user who last updated this row (foreign key to FND_LOGINS.LOGIN_ID). You should set this to NULL, or to 0 if NULL is not allowed

Since every table containing Who columns has several foreign keys to the tables FND\_USER and FND\_LOGINS, we do not include the foreign key columns LAST\_UPDATED\_BY, CREATED\_BY, or LAST\_UPDATE\_LOGIN in a table's list of foreign keys.

#### **Additional Who Columns for Concurrent Programs**

Some Oracle Product Development Exchange applications tables also contain several additional Who columns to distinguish between changes a user makes with a form and changes a concurrent program makes. When a concurrent program updates or inserts a row in a table, the concurrent program populates the following additional Who columns:

<b>REQUEST_ID</b>	Concurrent request ID of program that last updated this row (foreign key to FND_CONCURRENT_REQUESTS.REQUEST_ID)
<b>PROGRAM_APPLICATION_ID</b>	Application ID of program that last updated this row (foreign key to FND_APPLICATION.APPLICATION_ID)
<b>PROGRAM_ID</b>	Program ID of program that last updated this row (foreign key to FND_CONCURRENT_PROGRAM.CONCURRENT_PROGRAM_ID)
<b>PROGRAM_UPDATE_DATE</b>	Date when a program last updated this row

Since every table containing these additional Who columns has several foreign keys to the tables FND\_CONCURRENT\_REQUESTS, FND\_APPLICATION, and FND\_CONCURRENT\_PROGRAM, we do not include the foreign key columns REQUEST\_ID, PROGRAM\_APPLICATION\_ID, or PROGRAM\_ID in a table's list of foreign keys.



## Columns Reserved for Country-Specific Features

---

Some tables have GLOBAL\_ATTRIBUTE columns that support additional features added to Oracle Product Development Exchange to meet statutory requirements and common business practices in your country or region. For details on these columns, refer to the Appendix in *Oracle Financials Global Technical Reference Manual*. To read more about the features that these columns support, look for a User Guide appropriate to your country; for example, see the *Oracle Financials for the Czech Republic User Guide*.

---

## Indexes

If an Oracle Product Development Exchange applications table uses an Oracle8i index, we list the database columns that comprise that index, in sequential order.

**Note:** The indexes we document in this manual correspond to unique keys we specified during product development and testing. In some cases, we may add additional indexes during the porting process to fine-tune performance on specific platforms; therefore, there may be minor differences between the indexes documented in this book and the indexes for production versions of Oracle Product Development Exchange.

---

## Sequences

your Oracle Exchange application uses Oracle8i sequence generators to generate unique integers. If any table column gets its value from an Oracle8i sequence generator, we list the name of the corresponding sequence generator and the name of the column that stores the unique integer.

---

## Database Triggers

If a table has one or more active database triggers, we provide a brief explanation of each database trigger and when the trigger fires.



---

## View Derivation

For each Oracle Product Development Exchange applications view you may need to reference, we include important elements from the SQL statement that defines or creates a view. By studying this view definition, you can understand exactly how a view derives its contents.

## ENI\_CO\_HEADER\_V

This is the view for Change Order Headers. It is created from the following base tables: ipd\_change\_headers, fnd\_lookups, fnd\_lookups, fnd\_lookups, fnd\_lookups, fnd\_lookups and hz\_parties.

### View Definition

```
CREATE VIEW ENI_CO_HEADER_V
as SELECT
    ch.CHANGE_HEADER_ID, ch. CHANGE_NUMBER, ch.status_code, sc.meaning, ch.SCOPE_CODE,
    im.meaning, ch.PRIORITY_CODE, pc.meaning, ch.ASSIGNED_TO_PERSON_ID, per.party_name,
    nvl(ch.PHASE_WHEN_STARTED_CODE, 'NA'), nvl(pwsc.meaning, 'Unassigned'),
    nvl(ch.PHASE_WHEN_COMPLETED_CODE, 'NA'), nvl(pwcc.meaning, 'Unassigned'),
    trunc(ch.CREATION_DATE), to_char(ch.creation_date, 'fmMonth YYYY'),
    to_char(ch.creation_date, 'YYYY'), trunc(ch.IMPLEMENTATION_DATE),
    to_char(ch.implementation_date, 'fmMonth YYYY'), to_char(ch.implementation_date, 'YYYY'),
    ch.change_mgmt_type_code, ch.late_change, 1, ((decode(ch.implementation_date, null,
    trunc(ch.creation_date), trunc(ch.implementation_date)))-trunc(ch.creation_date)) from
    ipd_change_headers ch, fnd_lookups sc, fnd_lookups im, fnd_lookups pc, fnd_lookups pwcc,
    fnd_lookups pwsc, hz_parties per
    Where ch.status_code =sc.lookup_code and sc.lookup_type = 'IPD_CHANGE_MGMT_STATUS' and
    ch.scope_code =im.lookup_code and im.lookup_type = 'IPD_CHANGE_MGMT_SCOPE' and ch.priority_code
    =pc.lookup_code and pc.lookup_type = 'IPD_CHANGE_MGMT_PRIORITIES' and
    nvl(ch.PHASE_WHEN_STARTED_CODE, 'NA')=pwsc.lookup_code (+) and
    pwsc.lookup_type(+)='IPD_PROJECT_PHASES' and
    nvl(ch.PHASE_WHEN_COMPLETED_CODE, 'NA')=pwcc.lookup_code (+) and
    pwcc.lookup_type(+)='IPD_PROJECT_PHASES' and per.party_id=ch.ASSIGNED_TO_PERSON_ID
```

### Column Descriptions

Name	Null?	Type	Description
CHANGE_HEADER_ID	NOT NULL	NUMBER	Change Order/Request/Issue Identifier
CHANGE_NUMBER	NOT NULL	VARCHAR2(30)	Change Order/Request/Issue Number
STATUS_CODE	NOT NULL	VARCHAR2(30)	Status Code
STATUS_DESCRIPTION	NOT NULL	VARCHAR2(80)	Status Description
SCOPE_CODE	NOT NULL	VARCHAR2(30)	Scope Code
SCOPE_DESCRIPTION	NOT NULL	VARCHAR2(80)	Scope Description
PRIORITY_CODE	NOT NULL	VARCHAR2(30)	Priority Code
PRIORITY_DESCRIPTION	NOT NULL	VARCHAR2(80)	Priority Description
ASSIGNED_TO_PERSON_ID	NOT NULL	NUMBER	Assigned to Person Identifier
ASSIGNED_TO_PERSON_NAME	NOT NULL	VARCHAR2(255)	Assigned to Person Name
PHASE_STARTED_CODE	NULL	VARCHAR2(30)	Phase Started Code
PHASE_STARTED_DESC	NULL	VARCHAR2(80)	Phase Started Description
PHASE_COMPLETED_CODE	NULL	VARCHAR2(30)	Phase Completed Code
PHASE_COMPLETED_DESC	NULL	VARCHAR2(80)	Phase Completed Description
CREATION_DATE	NULL	DATE	Standard Who column
CREATION_MONTH	NULL	VARCHAR2(32)	Standard Who column (month)
CREATION_YEAR	NULL	VARCHAR2(4)	Standard Who column (year)
IMPLEMENTATION_DATE	NULL	DATE	Implementation Date
IMPLEMENTATION_MONTH	NULL	VARCHAR2(32)	Implementation Month
IMPLEMENTATION_YEAR	NULL	VARCHAR2(4)	Implementation Year
CHANGE_MGMT_TYPE	NOT NULL	VARCHAR2(30)	Change Management Type Code

Oracle Proprietary, Confidential Information--Use Restricted by Contract



LATE_CHANGE	NOT NULL	VARCHAR2(30)	Late Change
EXISTENCE	NULL	NUMBER	Used for counting calculations
CYCLE_TIME	NULL	NUMBER	Difference between IMPLEMENTATION_DATE and CREATION_DATE

---

## ENI\_MONTHS

ENI\_MONTH stores the last day of the calendar months from the year 1951 to 2010.

### *Column Descriptions*

<u>Name</u>	<u>Null?</u>	<u>Type</u>	<u>Description</u>
ID	NOT NULL	DATE	Unique Identifier for the last day of each calendar month

## ENI\_PRODUCT\_STRUCTURES\_V

This is the view for product structures. It is created from the following base tables and views: IPD\_PS\_COMPONENTS, IPD\_PRODUCT\_STRUCTURES, IPD\_PS\_REVISIONS, ENI\_MONTHS, ICX\_POR\_CATEGORY\_ITEMS, IPD\_ITEMS\_V, ICX\_POR\_CATEGORIES\_VL, ICX\_POR\_CATEGORY\_ITEMS, and ICX\_POR\_CATEGORIES\_VL.

### View Definition

```
CREATE VIEW ENI_PRODUCT_STRUCTURES_V
as SELECT
    a.product_structure_id, a.name, b.component_item_id,d.value, compitem.party_item_number,
    compcat.rt_category_id , compcat.category_name , a.parent_item_id, paritem.party_item_number,
    parcat.rt_category_id , parcat.category_name , b.effectivity_date , b.effectivity_date,
    b.effectivity_date , b.disable_date, b.disable_date,b.disable_date, 1, c.revision,
    c.effectivity_date, c.disable_date from ipd_ps_components b, ipd_product_structures a,
    ipd_ps_revisions c, eni_month d, icx_por_category_items compitemcat,ipd_items_v compitem,
    ICX_POR_CATEGORIES_VL compcat, icx_por_category_items paritemcat, ICX_POR_CATEGORIES_VL parcat,
    ipd_items_v paritem
    Where a.name is null and a.product_structure_id = b.product_structure_id and
    a.product_structure_id=c.product_structure_id and d.value between
    add_months(c.effectivity_date, -1) and nvl(c.disable_date, d.value+1) and ((d.value between
    add_months(b.effectivity_date, -1) and nvl(b.disable_date, d.value+1) and (b.effectivity_type
    is null) or (b.effectivity_type <>'PSREVISION')) or(b.effectivity_type ='PSREVISION'and
    c.revision between b.effective_from and nvl(b.effective_to, c.revision))) and
    compitemcat.rt_item_id = b.component_item_id and
    compitemcat.rt_category_id=compcat.rt_category_id and b.component_item_id=compitem.item_id and
    paritemcat.rt_item_id = a.parent_item_id and paritemcat.rt_category_id=parcat.rt_category_id
    and a.parent_item_id=paritem.item_id
```

### Column Descriptions

Name	Null?	Type	Description
PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product structure unique identifier
PS_NAME	NULL	VARCHAR2(30)	Product structure name
COMPONENT_ITEM_ID	NOT NULL	NUMBER	Component item identifier
MONTH	NULL	DATE	Month
COMPONENT_ITEM_NAME	NULL	VARCHAR2(700)	Component item name
COMP_ITEM_CATEGORY_ID	NOT NULL	NUMBER	Unique identifier for the component's category
COMP_ITEM_CATEGORY_NAME	NOT NULL	VARCHAR2(250)	Component item category name
PARENT_ITEM_ID	NOT NULL	NUMBER	Parent item identifier
PARENT_ITEM_NAME	NULL	VARCHAR2(700)	Parent item name
PARENT_ITEM_CATEGORY_ID	NOT NULL	NUMBER	Parent item category identifier
PARENT_ITEM_CATEGORY_NAME	NOT NULL	VARCHAR2(250)	Parent item category name
EFFECTIVITY_DAY	NOT NULL	DATE	Start effectivity day of the component
EFFECTIVITY_MONTH	NULL	DATE	Start effectivity month of the component
EFFECTIVITY_YEAR	NULL	DATE	Start effectivity year of the component



DISABLE_DAY	NULL	DATE	Date (day) when component is disabled
DISABLE_MONTH	NULL	DATE	Date (month) when component is disabled
DISABLE_YEAR	NULL	DATE	Date (year) when component is disabled
EXISTENCE	NULL	NUMBER	Used for counting calculations
REVISION	NULL	VARCHAR2(60)	Product structure revision name
REV_EFFECTIVITY_DAY	NOT NULL	DATE	Product structure revision start effectivity date
REV_DISABLE_DAY	NULL	DATE	Product structure revision disable date

---

## ENI\_PS\_COMPLEXITY\_V

This is the complexity measure view created from the view  
ENI\_PRODUCT\_STRUCTURES\_V.

### View Definition

```
CREATE VIEW ENI_PS_COMPLEXITY_V
as SELECT
    a.parent_item_id, a.parent_item_id, a.parent_item_name, last_day(a.month) eff_month,
    to_char(a.month, 'fmMonth yyyy'), trunc(a.month, 'YYYY'), to_char(a.month, 'YYYY'),
    a.parent_item_category_id, a.parent_item_category_name, sum(existence), a.revision from
    eni_product_structures_v a where last_day(a.month) >= a.effectivity_day and
    last_day(a.month) < nvl(a.disable_day, last_day(a.month) + 1) and last_day(a.month) >=
    a.rev_effectivity_day and last_day(a.month) < nvl(a.rev_disable_day, last_day(a.month)+1)
group by a.parent_item_id, a.month, parent_item_category_id, a.parent_item_category_name,
a.parent_item_name, a.revision
```

### Column Descriptions

Name	Null?	Type	Description
ITEM_ID	NOT NULL	NUMBER	Unique Identifier for Item
RT_ITEM_ID	NOT NULL	NUMBER	Unique Identifier for Item (same as ITEM_ID)
ITEM_NAME	NULL	VARCHAR2(700)	Item Name
EFF_MONTH	NULL	DATE	Month for which this complexity measure is valid for this item
EFF_MONTH_DP	NULL	VARCHAR2(32)	Start Effectivity Month of the component (formatted for display purposes)
EFF_YEAR	NULL	DATE	Year for which this complexity measure is valid for this item
EFF_YEAR_DP	NULL	VARCHAR2(4)	Start Effectivity Year of the component (formatted for display purposes)
CAT_ID	NOT NULL	NUMBER	Unique identifier for the Item's category
CAT_NAME	NOT NULL	VARCHAR2(250)	Item's category name
COMPLEXITY	NULL	NUMBER	Complexity measure
REVISION	NULL	VARCHAR2(60)	Product structure revision name

## ENI\_PS\_REUSE\_V

This is the reuse measure view created from the view  
ENI\_PRODUCT\_STRUCTURES\_V.

### View Definition

```
CREATE VIEW ENI_PS_REUSE_V
as SELECT
    a.component_item_id item_id, a.component_item_id ,a.component_item_name item_name,
    last_day(a.month) eff_month, to_char(a.month, 'fmMonth yyyy'), trunc(a.month,'YYYY'),
    to_char(a.month, 'YYYY'), a.comp_item_category_id, a.comp_item_category_name cat_name,
    sum(existence) reuse, a.revision from eni_product_structures_v a where last_day(a.month) >=
    a.effectivity_day and last_day(a.month) < nvl(a.disable_day, last_day(a.month)+1) and
    last_day(a.month) >= a.rev_effectivity_day and last_day(a.month) < nvl(a.rev_disable_day,
    last_day(a.month)+1) group by a.component_item_id, comp_item_category_id,
    a.comp_item_category_name, a.component_item_name, a.month, a.revision
```

### Column Descriptions

Name	Null?	Type	Description
ITEM_ID	NOT NULL	NUMBER	Unique Identifier for Item
RT_ITEM_ID	NOT NULL	NUMBER	Unique Identifier for Item (same as ITEM_ID)
ITEM_NAME	NULL	VARCHAR2(700)	Item Name
EFF_MONTH	NULL	DATE	Month for which this reuse measure is valid for this item
EFF_MONTH_DP	NULL	VARCHAR2(32)	Start Effectivity Month of the component (formatted for display purposes)
EFF_YEAR	NULL	DATE	Year for which this complexity measure is valid for this item
EFF_YEAR_DP	NULL	VARCHAR2(4)	Start Effectivity Year of the component (formatted for display purposes)
CAT_ID	NOT NULL	NUMBER	Unique identifier for the Item's category
CAT_NAME	NOT NULL	VARCHAR2(250)	Item's category name
REUSE	NULL	NUMBER	Reuse measure
REVISION	NULL	VARCHAR2(60)	Product structure revision name

## ICX\_POR\_ITEMS

Stores information and root descriptors for non-translatable data for each item in the catalog. Root descriptor columns are in the format Axxxx where xxxx is an integer. Some of the root descriptors are seeded by Oracle while others may be generated through the use of the application.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
HZ_PARTIES	PARTY_ID	SUPPLIER_ID

### Column Descriptions

Name	Null?	Type	Description
RT_ITEM_ID	NOT NULL	NUMBER	Internally generated Item identifier
ORC_ITEM_ID	NULL	NUMBER	Item unique identifier in Oracle (IP only)
ORC_ITEM_NUM	NULL	VARCHAR2(700)	Item number in Oracle (IP only)
TITLE	NULL	VARCHAR2(500)	ID of the Originating system data source/publishing entity set to 'Oracle' for Oracle-sourced data
SUPPLIER_ID	NOT NULL	NUMBER	Unique identifier for the supplier owning the item.
BATCH_JOB_NUM	NULL	NUMBER	Batch job number that created/updated the item
QUERY	NULL	VARCHAR2(240)	Not used
TYPE	NULL	NUMBER	Not used
A1	NULL	VARCHAR2(700)	Reserved for Supplier name
A2	NULL	VARCHAR2(700)	Reserved for Supplier ID
A3	NULL	VARCHAR2(700)	Reserved for Supplier Item Number
A4	NULL	VARCHAR2(700)	Reserved for Manufacturer (Not used in WR4)
A5	NULL	VARCHAR2(700)	Reserved for Manufacturer Part Number (Not used in WR4)
A10	NULL	NUMBER	Not used
A11	NULL	VARCHAR2(700)	Not used
A12	NULL	VARCHAR2(700)	Reserved for Internal Item Number (IP only)
A13	NULL	VARCHAR2(700)	Reserved for Picture or Image file name (PICTURE)
A14	NULL	VARCHAR2(700)	Reserved for Picture or image URL (PICTURE_URL)
A15	NULL	VARCHAR2(700)	Reserved for Contract Number (IP only)
A16	NULL	VARCHAR2(700)	Reserved for Contract Line (IP only)
A17	NULL	VARCHAR2(700)	Reserved for Contract Price (IP only)
A18	NULL	VARCHAR2(700)	Reserved for Contract Currency (IP only)
A19	NULL	VARCHAR2(700)	Reserved for Contract Rate Type (IP only)
A20	NULL	VARCHAR2(700)	Reserved for Contract Rate Date (IP Only)

A21	NULL	VARCHAR2(700)	Reserved for Contract Rate (IP only)
A22	NULL	VARCHAR2(700)	Reserved for attachment URL (ATTACHMENT_URL)
A24	NULL	VARCHAR2(700)	Reserved for UNSPSC code (UNSPSC)
A25	NULL	VARCHAR2(700)	Reserved for Availability
A26	NULL	NUMBER	Reserved for order lead time (LEAD_TIME). specified in days
A27	NULL	VARCHAR2(700)	Reserved for Functional Currency Price (IP only)
A28	NULL	VARCHAR2(700)	Reserved for Functional Currency (IP only)
A29	NULL	VARCHAR2(700)	Reserved for item type (ITEM_TYPE)
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column
REQUEST_ID	NULL	NUMBER	Extended Who Column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Extended Who Column
PROGRAM_ID	NULL	NUMBER	Extended Who Column
PROGRAM_UPDATE_DATE	NULL	DATE	Extended Who Column
A30	NULL	VARCHAR2(700)	Reserved for sellable item (SELLABLE_ITEM). Product Development specific attribute. If set to N, the item will not be visible in the standard catalog.
A31	NULL	VARCHAR2(700)	Reserved for development item (DEVELOPMENT_ITEM). Product development specific attribute.
A32	NULL	NUMBER	Reserved for Owner Party Id. Product development specific attribute. Should be populated if A31 is Yes.
A33	NULL	VARCHAR2(700)	Reserved for Minimum Inventory Quantity
A34	NULL	VARCHAR2(700)	Reserved for Maximum Inventory Quantity
A35	NULL	VARCHAR2(700)	Reserved for pricing attributes.
A36	NULL	VARCHAR2(700)	Reserved for pricing attributes.
ACTION	NULL	VARCHAR2(30)	Action last taken on the item. Possible values are ADD and UPDATE.
STATUS	NULL	VARCHAR2(30)	Status of the item relative to publication. Possible values are PUBLISHED and UNPUBLISHED.

*Indexes*

Index Name	Index Type	Sequence	Column Name
ICX_POR_ITEMS_N1	NOT UNIQUE	5	ORC_ITEM_ID
ICX_POR_ITEMS_N2	NOT UNIQUE	5	A3
ICX_POR_ITEMS_N3	NOT UNIQUE	5	A1
ICX_POR_ITEMS_N4	NOT UNIQUE	5	SUPPLIER_ID
ICX_POR_ITEMS_N5	NOT UNIQUE	5	STATUS
ICX_POR_ITEMS_N6	NOT UNIQUE	5	BATCH_JOB_NUM
ICX_POR_ITEMS_U1	UNIQUE	5	RT_ITEM_ID

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**



ICX_POR_ITEMS_U2	UNIQUE	5	A1
		10	A3
		15	STATUS

## IPD\_CHANGE\_HEADERS

IPD\_CHANGE\_HEADERS contains information about change orders, change requests and issues. You need one row for each change order, change request, or issue that you create.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_TYPES	TYPE_ID	CHANGE_HEADER_TYPE_ID
IPD_ITEM_REVISIONS_B	REVISION_ID	ITEM_REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
CHANGE_HEADER_ID	NOT NULL	NUMBER	Change Order/Request/Issue Identifier
CHANGE_NUMBER	NOT NULL	VARCHAR2(30)	Change Order/Request/Issue Number
CREATION_DATE	NOT NULL	DATE	Standard Who column
CREATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
PROJECT_ID	NOT NULL	NUMBER	Project Identifier
TASK_ID	NOT NULL	NUMBER	Task Identifier
CHANGE_MGMT_TYPE_CODE	NOT NULL	VARCHAR2(30)	Change Management Type Code
CHANGE_HEADER_TYPE_ID	NOT NULL	NUMBER	Change Type Identifier
WORKFLOW_PROCESS	NOT NULL	VARCHAR2(30)	Workflow Process
REQUESTOR_ID	NULL	NUMBER	Requestor Identifier
SCHEDULE_DATE	NULL	DATE	Scheduled Completion Date
IMPLEMENTATION_DATE	NULL	DATE	Implementation Date
APPROVAL_DATE	NULL	DATE	Approval Date
CANCELLATION_DATE	NULL	DATE	Cancellation Date
APPROVAL_REQUEST_DATE	NULL	DATE	Approval Request Date
CANCELLATION_COMMENTS	NULL	VARCHAR2(240)	Cancellation Comments
ITEM_ID	NULL	NUMBER	Item Identifier
STATUS_CODE	NOT NULL	VARCHAR2(30)	Status Code
PRIORITY_CODE	NOT NULL	VARCHAR2(30)	Priority Code
SCOPE_CODE	NOT NULL	VARCHAR2(30)	Scope Code
OTHER_COST	NULL	NUMBER	Other Cost
CURRENCY_CODE	NULL	VARCHAR2(15)	Currency Code
EFFORT	NULL	NUMBER	Effort
DURATION	NULL	NUMBER	Duration
PHASE_WHEN_STARTED_CODE	NULL	VARCHAR2(30)	Phase Started Code
PHASE_WHEN_COMPLETED_CODE	NULL	VARCHAR2(30)	Phase Completed Code
ASSIGNED_TO_PERSON_ID	NOT NULL	NUMBER	Assigned To Person Identifier
LATE_CHANGE	NOT NULL	VARCHAR2(30)	Late Change
REQUEST_ID	NULL	NUMBER	Concurrent Request Identifier
PROGRAM_ID	NULL	NUMBER	Program Identifier
PROGRAM_APPLICATION_ID	NULL	NUMBER	Program Application Identifier
PROGRAM_UPDATE_DATE	NULL	DATE	Standard Who column
REASON_CODE	NULL	VARCHAR2(30)	Reason Code
IMPLEMENTATION_COST	NULL	NUMBER	Implementation Cost
PARENT_ITEM_ID	NULL	NUMBER	Parent Item Identifier
TOP_LEVEL_ITEM_ID	NULL	NUMBER	Top Level Item Identifier
DATE_COMMITTED	NULL	DATE	Date Committed

Oracle Proprietary, Confidential Information--Use Restricted by Contract

DATE_REQUIRED	NULL	DATE	Date Required
ITEM_REVISION_ID	NULL	NUMBER	Item Revision Identifier
PROGRESS_STATUS_CODE	NOT NULL	VARCHAR2(30)	Progress Status Code

*Indexes*

<u>Index Name</u>	<u>Index Type</u>	<u>Sequence</u>	<u>Column Name</u>
IPD_CHANGE_HEADERS_N1	NOT UNIQUE	1	CHANGE_MGMT_TYPE_CODE
		2	ASSIGNED_TO_PERSON_ID
IPD_CHANGE_HEADERS_N2	NOT UNIQUE	1	CHANGE_MGMT_TYPE_CODE
		2	ITEM_ID
		3	ITEM_REVISION_ID
IPD_CHANGE_HEADERS_N3	NOT UNIQUE	1	CHANGE_MGMT_TYPE_CODE
		2	PROJECT_ID
		3	TASK_ID
IPD_CHANGE_HEADERS_U1	UNIQUE	1	CHANGE_HEADER_ID
IPD_CHANGE_HEADERS_U2	UNIQUE	1	CHANGE_MGMT_TYPE_CODE
		2	CHANGE_NUMBER

---

## IPD\_CHANGE\_HEADERS\_TL

IPD\_CHANGE\_HEADERS\_TL contains information about change orders, change requests and issues.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_HEADERS	CHANGE_HEADER_ID	CHANGE_HEADER_ID

### Column Descriptions

Name	Null?	Type	Description
CHANGE_HEADER_ID	NOT NULL	NUMBER	Change Order/Request/Issue Identifier
LANGUAGE	NOT NULL	VARCHAR2(4)	Language
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source Language
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
SHORT_DESCRIPTION	NOT NULL	VARCHAR2(240)	Short Description
LONG_DESCRIPTION	NULL	VARCHAR2(4000)	Long Description
RESOLUTION	NULL	VARCHAR2(4000)	Resolution

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_HEADERS_TL_U1	UNIQUE	10	CHANGE_HEADER_ID
		20	LANGUAGE

## IPD\_CHANGE\_LINES

IPD\_CHANGE\_LINES stores data for change lines. Each row in this table represents a specific change line and is uniquely identified by its CHANGE\_LINE\_ID. A change line's change type is determined by its CHANGE\_LINE\_TYPE\_ID, which references a row in the IPD\_CHANGE\_TYPES table. A change line's subject can be one of the following objects: item, product structure, project, task, or document. Therefore, a line's SUBJECT\_ID is an identifier for one of these objects, depending on the subject type. A line's subject type is determined by the SUBJECT\_TYPE\_CODE of its line type in IPD\_CHANGE\_TYPES. Each row in the IPD\_CHANGE\_LINES table is a child to a change order, change request or issue in the IPD\_CHANGE\_HEADERS table and references its parent by its CHANGE\_HEADER\_ID. LINE\_NUMBER is unique within a CHANGE\_HEADER\_ID.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_HEADERS	CHANGE_HEADER_ID	CHANGE_HEADER_ID
IPD_CHANGE_TYPES	TYPE_ID	CHANGE_LINE_TYPE_ID

### Column Descriptions

Name	Null?	Type	Description
CHANGE_LINE_ID (PK)	NOT NULL	NUMBER	Change line unique identifier
CHANGE_HEADER_ID	NOT NULL	NUMBER	Change header identifier
CHANGE_LINE_TYPE_ID	NOT NULL	NUMBER	Change type identifier
LINE_NUMBER	NOT NULL	NUMBER	Line number within a change header
SUBJECT_ID	NOT NULL	NUMBER	Subject identifier - Could be one of the following - Item id, Product Structure Id, Project Id, Task Id, Fnd Document Id
SUBJECT_NEW_REVISION	NULL	VARCHAR2(54)	New Revision for Subject
SUBJECT_CURRENT_REVISION_ID	NULL	NUMBER	Current revision of Subject for which this change is being created. Relevant for the following subject types: Item, Product Structure, Document
CREATION_DATE	NOT NULL	DATE	Standard Who column
CREATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
WORKFLOW_PROCESS	NULL	VARCHAR2(30)	Name of Workflow process for the change line
PRIORITY_CODE	NULL	VARCHAR2(30)	Priority code
SCOPE_CODE	NULL	VARCHAR2(30)	Scope code
IMPLEMENTATION_COST	NULL	NUMBER	Implementation Cost
OTHER_COST	NULL	NUMBER	Other cost
EFFORT	NULL	NUMBER	Effort in days
DURATION	NULL	NUMBER	Duration in days
APPROVAL_DATE	NULL	DATE	Approval date
SCHEDULE_DATE	NULL	DATE	Schedule date
IMPLEMENTATION_DATE	NULL	DATE	Implementation date

Oracle Proprietary, Confidential Information--Use Restricted by Contract



STATUS_CODE	NULL	VARCHAR2(30)	Status code
REQUEST_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who column
CURRENCY_CODE	NULL	VARCHAR2(15)	Currency code

*Indexes*

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_LINES_N1	NOT UNIQUE	1	CHANGE_HEADER_ID
		2	SUBJECT_ID
		3	CHANGE_LINE_TYPE_ID
IPD_CHANGE_LINES_U1	UNIQUE	1	CHANGE_LINE_ID
IPD_CHANGE_LINES_U2	UNIQUE	1	CHANGE_HEADER_ID
		2	LINE_NUMBER



---

## IPD\_CHANGE\_LINES\_TL

IPD\_CHANGE\_LINES\_TL is a translation table that stores the language-specific description for a change line.

### Column Descriptions

Name	Null?	Type	Description
CHANGE_LINE_ID	NOT NULL	NUMBER	Change line unique identifier
LANGUAGE	NOT NULL	VARCHAR2(4)	Language
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source language
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LINE_DESCRIPTION	NOT NULL	VARCHAR2(4000)	Line description

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_LINES_TL_U1	UNIQUE	10	CHANGE_LINE_ID
		20	LANGUAGE

## IPD\_CHANGE\_REVISIONS

IPD\_CHANGE\_REVISIONS is a table that stores data for change revisions. Each row in this table represents a specific change revision and is uniquely identified by its CHANGE\_REVISION\_ID.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_HEADERS	CHANGE_HEADER_ID	CHANGE_HEADER_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID (PK)	NOT NULL	NUMBER	Change Revision Identifier
CHANGE_HEADER_ID	NOT NULL	NUMBER	Change Header Identifier
REVISION	NOT NULL	VARCHAR2(54)	Change revision
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
EFFECTIVITY_DATE	NOT NULL	DATE	Effective date of the change revision
DISABLE_DATE	NULL	DATE	Disable date of the change revision
REVISION_SEGMENT1	NOT NULL	VARCHAR2(10)	Key flexfield for storing revision segment
REVISION_SEGMENT2	NULL	VARCHAR2(10)	Key flexfield for storing revision segment
REVISION_SEGMENT3	NULL	VARCHAR2(10)	Key flexfield for storing revision segment
REVISION_SEGMENT4	NULL	VARCHAR2(10)	Key flexfield for storing revision segment
REVISION_SEGMENT5	NULL	VARCHAR2(10)	Key flexfield for storing revision segment
REVISION_SORT_SEQ_NUM	NULL	NUMBER	Sequence number for the change revision
REVISED_FROM_REVISION_ID	NULL	NUMBER	Revision identifier of the change revision from which the current revision was revised
STATUS	NULL	VARCHAR2(30)	Change revision status
PRIMARY_REASON_CODE	NULL	VARCHAR2(30)	Primary reason code
REQUEST_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who column
DATE_COMMITTED	NULL	DATE	Committed date of the change revision

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_REVISIONS_U1	UNIQUE	1	REVISION_ID
IPD_CHANGE_REVISIONS_U2	UNIQUE	1	CHANGE_HEADER_ID
		2	REVISION



---

## IPD\_CHANGE\_REVISIONS\_TL

IPD\_CHANGE\_REVISIONS is a table that stores data for change revisions. Each row in this table represents a specific change revision and is uniquely identified by its CHANGE\_REVISION\_ID.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_REVISIONS	REVISION_ID	REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID (PK)	NOT NULL	NUMBER	Change revision identifier
LANGUAGE (PK)	NOT NULL	VARCHAR2(4)	Language for the change revision description
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Stores Language for user's environment.
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
DESCRIPTION	NULL	VARCHAR2(2000)	Language dependent detailed description for the change revision
SHORT_DESCRIPTION	NULL	VARCHAR2(240)	Language dependent short description for the change revision

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_REVISIONS_TL_U1	UNIQUE	10	REVISION_ID
		20	LANGUAGE

## IPD\_CHANGE\_SUBMIT\_REVISIONS

IPD\_CHANGE\_SUBMIT\_REVISIONS is a table that stores information about the number of submitted Workflow processes for a particular change header or change line. Each row is associated with a submitted Workflow process. The OBJECT\_TYPE can be "HEADER" or "LINE." OBJECT\_ID is either a header's Change Header ID from the IPD\_CHANGE\_HEADERS table, or a change line's Change Line ID from the IPD\_CHANGE\_LINES table. REVISION\_NUMBER is a count that keeps track of the number of submitted processes for a particular object.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_HEADERS	CHANGE_HEADER_ID	OBJECT_ID
IPD_CHANGE_HEADERS	CHANGE_HEADER_ID	OBJECT_ID
IPD_CHANGE_LINES	CHANGE_LINE_ID	OBJECT_ID
IPD_CHANGE_LINES	CHANGE_LINE_ID	OBJECT_ID

### Column Descriptions

Name	Null?	Type	Description
OBJECT_CODE (PK)	NOT NULL	VARCHAR2(30)	Object Code - HEADER or LINE
OBJECT_ID (PK)	NOT NULL	NUMBER	Object ID - Change Header ID or Change Line ID, depending on the OBJECT_CODE
REVISION_NUMBER (PK)	NOT NULL	NUMBER	Revision number
PROCESS_NAME	NOT NULL	VARCHAR2(30)	Workflow process
SUBMIT_DATE	NOT NULL	DATE	Date Workflow process is submitted
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
CREATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column
REQUEST_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_ID	NULL	NUMBER	Concurrent Who column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_SUBMIT_REVISIONS_N1	NOT UNIQUE	1	OBJECT_CODE
		2	OBJECT_ID
		3	REVISION_NUMBER

## IPD\_CHANGE\_TYPES

IPD\_CHANGE\_TYPES is used by Change Management to store Change Types. You need a single row for each type you create. Each header type here has a management type code that is one of following: change order, change request, issue. A header type also has change number prefix. Line Types on the other hand have subject type codes associated with them.

### Column Descriptions

Name	Null?	Type	Description
TYPE_ID (PK)	NOT NULL	NUMBER	Change Type Identifier
CHANGE_TYPE_CODE	NOT NULL	VARCHAR2(30)	Code of the Type. Could be HEADER or LINE
CHANGE_TYPE_NAME	NOT NULL	VARCHAR2(30)	the name of the Type. This will be unique with in the type code.
CHANGE_MANAGEMENT_TYPE_CODE	NULL	VARCHAR2(30)	Denotes the Management type code for the HEADER types.Values could be one of following: Change Order, change Request, Issue.
SUBJECT_TYPE_CODE	NULL	VARCHAR2(30)	Denotes the subject type for change line. Applies on ly to the change type code of LINE.Values could be one of following: Item, Project, Task, Document, Product Structure, Text.
WORKFLOW_PROCESS	NULL	VARCHAR2(30)	What default work flow Process your type wants to use. User can always change this whenthey use this type.
CHANGE_NUMBER_PREFIX	NULL	VARCHAR2(3)	3 character prefix for the Header Types only.
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_TYPES_U1	UNIQUE	1	TYPE_ID
IPD_CHANGE_TYPES_U2	UNIQUE	3	CHANGE_MANAGEMENT_TYPE_CODE
		4	CHANGE_TYPE_NAME
		5	CHANGE_TYPE_CODE

---

## IPD\_CHANGE\_TYPES\_TL

IPD\_CHANGE\_TYPES\_TL is a translation table that stores the language-specific description for a change type.

### Column Descriptions

Name	Null?	Type	Description
TYPE_ID (PK)	NOT NULL	NUMBER	Change Type Identifier
LANGUAGE (PK)	NOT NULL	VARCHAR2(4)	Language for the Change Type Description.
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Stores Language for user's environment.
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
DESCRIPTION	NULL	VARCHAR2(2000)	Language dependent description for the Change Type.

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_CHANGE_TYPES_TL_U1	UNIQUE	10 20	TYPE_ID LANGUAGE

## IPD\_CONFERENCE\_HEADERS

IPD\_CONFERENCE\_HEADERS contains information about conferences.

### Column Descriptions

Name	Null?	Type	Description
CONFERENCE_ID	NOT NULL	NUMBER	Conference Identifier
CONFERENCE_TITLE	NULL	VARCHAR2(240)	Conference Title
OBJECT_TYPE	NULL	VARCHAR2(30)	Object Type
OBJECT_ID1	NULL	NUMBER	Object identification
OBJECT_ID2	NULL	NUMBER	Object identification
WORKFLOW_PROCESS	NULL	VARCHAR2(30)	Workflow Process
ORGANIZER_ID	NOT NULL	NUMBER	Organizer identification
SCHEDULE_DATE	NOT NULL	DATE	The date when the conference will be held
DURATION	NOT NULL	NUMBER	How long the conference will be held
NO_OF_ATTENDEES	NULL	NUMBER	Number of people who attended the conference
FREQUENCY	NOT NULL	VARCHAR2(30)	Frequency of the conference
WEB_URL	NULL	VARCHAR2(300)	URL of the conference
WEB_USER_NAME	NULL	VARCHAR2(30)	Username to use when accessing the conference
WEB_PASSWORD	NULL	VARCHAR2(30)	Password to use when accessing the conference
PHONE_NUMBER	NULL	VARCHAR2(12)	Phone number to call when joining the conference
PHONE_MEETING_ID	NULL	NUMBER	Meeting identification of the conference
PHONE_PASS_CODE	NULL	VARCHAR2(30)	Passcode used in the conference
CREATION_DATE	NOT NULL	DATE	Standard Who Column
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column
REQUEST_ID	NULL	NUMBER	Concurrent Request Identifier
PROGRAM_ID	NULL	NUMBER	Program Identifier
PROGRAM_APPLICATION_ID	NULL	NUMBER	Program Application Identifier
PROGRAM_UPDATE_DATE	NULL	DATE	Program Application Update Date

---

## IPD\_EVENTS

IPD\_EVENTS stores event information. Each row in this table identifies a unique event, and is identified by EVENT\_ID. Each event belong to one of the event types.

### Column Descriptions

Name	Null?	Type	Description
EVENT_ID	NOT NULL	NUMBER	Event Identifier
EVENT_TYPE	NOT NULL	VARCHAR2(240)	Event type
OBJECT_NAME	NOT NULL	VARCHAR2(30)	object name
OBJECT_INSTANCE_KEY	NOT NULL	NUMBER	object instance
OBJECT_INSTANCE_USER_KEY	NOT NULL	VARCHAR2(240)	User name of the Object instance
OBJECT_INSTANCE_DESC_SQL	NOT NULL	VARCHAR2(1000)	SQL statement to get object instance information to construct the message
CHANGED_ATTRIBUTES	NOT NULL	VARCHAR2(2000)	Event attribute names
CHANGED_ATTRIBUTE_VALUES	NOT NULL	VARCHAR2(4000)	Event attribute values
OBJECT_INSTANCE_URL	NOT NULL	VARCHAR2(500)	URL to access object instance details
CHANGED_BY_PARTY_ID	NOT NULL	NUMBER	Party id of the person who did the changes
CHANGED_ON	NOT NULL	DATE	Date changed on
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column



---

## IPD\_EVENT\_NOTIFICATIONS

IPD\_EVENT\_NOTIFICATIONS stores event notifications sent to the subscribers. Each row in this table identifies a unique event notification, and is identified by NOTIFICATION\_ID.

*Column Descriptions*

Name	Null?	Type	Description
NOTIFICATION_ID (PK)	NOT NULL	NUMBER	Notification Identifier
EVENT_ID	NOT NULL	NUMBER	Event Identifier
RECIPIENT_PARTY_ID	NOT NULL	NUMBER	Recipient Party Identifier
SUBSCRIBER_PARTY_ID	NOT NULL	NUMBER	Subscriber Party Identifier
PRIORITY	NOT NULL	NUMBER	Priority of the Notification

---

## IPD\_EVENT\_SUBSCRIPTIONS

IPD\_EVENT\_SUBSCRIPTIONS stores subscription information. Each row in this table identifies a unique event subscription, and is identified by SUBSCRIPTION\_ID. Each subscription includes the: event type, subscriber identified by party\_id, object name, object instance and other attributes.

### Column Descriptions

Name	Null?	Type	Description
SUBSCRIPTION_ID	NOT NULL	NUMBER	Event subscription Identifier
EVENT_TYPE	NOT NULL	VARCHAR2(240)	Name of the Event type
OBJECT_NAME	NOT NULL	VARCHAR2(30)	Object name of the subscription
OBJECT_KEY	NOT NULL	NUMBER	Object key of the subscription
OBJECT_KEY_TYPE	NOT NULL	VARCHAR2(30)	Object key type
PARTY_ID	NOT NULL	NUMBER	Subscriber's Party id
EMAIL_FLAG	NOT NULL	VARCHAR2(1)	If EMAIL_FLAG = 'Y' , subscriber can receive notification by email, other wise, subscriber would not receive email
START_DATE	NOT NULL	DATE	Start date of the subscription
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
END_DATE	NULL	DATE	End date of the subscription
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_EVENT_SUBSCRIPTIONS_U1	UNIQUE	1	SUBSCRIPTION_ID

---

## IPD\_EVENT\_TYPES\_B

IPD\_EVENT\_TYPES\_B stores event type information. Each row in this table identifies a unique event type, and is identified by GUID. Events are grouped into different event types based on object name and change information.

### Column Descriptions

Name	Null?	Type	Description
GUID	NOT NULL	RAW	Unique event type Identifier
NAME	NOT NULL	VARCHAR2(240)	Event type name. The naming standard for Event Name is the same as that used for Java classes (for example, oracle.apps.item.DocumentListModified). Event names are case-sensitive
STATUS	NOT NULL	VARCHAR2(8)	Event Status( ACTIVE or INACTIVE)
OBJECT_NAME	NOT NULL	VARCHAR2(30)	Object Name
PRIVILEGE_NAME	NOT NULL	VARCHAR2(30)	Privilege name associated with the event type
OBJECT_INSTANCE_SQL_TEMPLATE	NOT NULL	VARCHAR2(1000)	Object instance SQL template, a query template to get object instance information
OBJECT_INSTANCE_URL_TEMPLATE	NOT NULL	VARCHAR2(240)	Object instance URL template, an URL template to access object instance detail web page
MESSAGE_GENERATE_FUNCTION	NOT NULL	VARCHAR2(61)	PL/SQL procedure name which will be dynamically
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_EVENT_TYPES_U1	UNIQUE	1	NAME

---

## IPD\_EVENT\_TYPES\_TL

IPD\_EVENT\_TYPES\_TL stores translated information for event types in IPD\_EVENT\_TYPES\_B. Each row includes the language the row is translated to, the translated attributes of the event type, and event type identifier (GUID) that uniquely identifies the event type.

### Column Descriptions

Name	Null?	Type	Description
GUID	NOT NULL	RAW	Unique event type Identifier
DISPLAY_NAME	NOT NULL	VARCHAR2(80)	Display name for the event type
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LANGUAGE	NOT NULL	VARCHAR2(4)	Language code
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source language codee
DESCRIPTION	NULL	VARCHAR2(4000)	Description for event type
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column



---

## IPD\_FND\_DOCUMENTS\_V

IPD\_FND\_DOCUMENTS\_V is view on IPD\_FND\_DOCUMENTS\_V,  
and provides aliases for doc\_attribute columns used in PDX.

### View Definition

```
CREATE VIEW IPD_FND_DOCUMENTS_V
  as SELECT
    ROW_ID,
  DOCUMENT_ID,
  CREATION_DATE,
  CREATED_BY,
  LAST_UPDATE_DATE,
  LAST_UPDATED_BY,
  LAST_UPDATE_LOGIN,
  DATATYPE_ID,
  DATATYPE_NAME,
  DESCRIPTION,
  FILE_NAME,
  MEDIA_ID,
  STORAGE_TYPE,
  CATEGORY_APPLICATION_ID,
  CATEGORY_ID,
  CATEGORY_DESCRIPTION,
  SECURITY_TYPE,
  SECURITY_ID,
  PUBLISH_FLAG,
  IMAGE_TYPE,
  USAGE_TYPE,
  USAGE_TYPE_DESCR,
  START_DATE_ACTIVE,
  END_DATE_ACTIVE,
  REQUEST_ID,
  PROGRAM_APPLICATION_ID,
  PROGRAM_ID,
  PROGRAM_UPDATE_DATE,
  DOC_ATTRIBUTE_CATEGORY,
  DOC_ATTRIBUTE1 REVISION,
  DOC_ATTRIBUTE2 REVISION_LABEL,
  DOC_ATTRIBUTE3 DOCUMENT_WF_KEY ,
  DOC_ATTRIBUTE4,
  DOC_ATTRIBUTE5,
  DOC_ATTRIBUTE6,
  DOC_ATTRIBUTE7,
  DOC_ATTRIBUTE8,
  DOC_ATTRIBUTE9,
  DOC_ATTRIBUTE10,
  DOC_ATTRIBUTE11,
  DOC_ATTRIBUTE12,
  DOC_ATTRIBUTE13,
  decode(datatype_id,7,doc_attribute14,'FND') orig_system,
  decode(datatype_id,7,doc_attribute15,document_id) orig_system_id
FROM fnd_documents_v1
```

Column Descriptions

Name	Null?	Type	Description
ROW_ID	NULL	ROWID	Refer to base table.
DOCUMENT_ID	NOT NULL	NUMBER	Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER	Refer to base table.
DATATYPE_ID	NOT NULL	NUMBER	Refer to base table.
DATATYPE_NAME	NOT NULL	VARCHAR2(30)	Refer to base table.
DESCRIPTION	NULL	VARCHAR2(255)	Refer to base table.
FILE_NAME	NULL	VARCHAR2(255)	Refer to base table.
MEDIA_ID	NULL	NUMBER	Refer to base table.
STORAGE_TYPE	NULL	NUMBER	Refer to base table.
CATEGORY_APPLICATION_ID	NULL	NUMBER	Refer to base table.
CATEGORY_ID	NOT NULL	NUMBER	Refer to base table.
CATEGORY_DESCRIPTION	NOT NULL	VARCHAR2(255)	Refer to base table.
SECURITY_TYPE	NOT NULL	NUMBER	Refer to base table.
SECURITY_ID	NULL	NUMBER	Refer to base table.
PUBLISH_FLAG	NOT NULL	VARCHAR2(1)	Refer to base table.
IMAGE_TYPE	NULL	VARCHAR2(10)	Refer to base table.
USAGE_TYPE	NULL	VARCHAR2(1)	Refer to base table.
USAGE_TYPE_DESCR	NOT NULL	VARCHAR2(80)	Refer to base table.
START_DATE_ACTIVE	NULL	DATE	Refer to base table.
END_DATE_ACTIVE	NULL	DATE	Refer to base table.
REQUEST_ID	NULL	NUMBER	Refer to base table.
PROGRAM_APPLICATION_ID	NULL	NUMBER	Refer to base table.
PROGRAM_ID	NULL	NUMBER	Refer to base table.
PROGRAM_UPDATE_DATE	NULL	DATE	Refer to base table.
DOC_ATTRIBUTE_CATEGORY	NULL	VARCHAR2(30)	Refer to base table.
REVISION	NULL	VARCHAR2(150)	Refer to base table.
REVISION_LABEL	NULL	VARCHAR2(150)	Refer to base table.
DOCUMENT_WF_KEY	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE4	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE5	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE6	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE7	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE8	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE9	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE10	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE11	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE12	NULL	VARCHAR2(150)	Refer to base table.
DOC_ATTRIBUTE13	NULL	VARCHAR2(150)	Refer to base table.
ORIG_SYSTEM	NULL	VARCHAR2(150)	Refer to base table.
ORIG_SYSTEM_ID	NULL	VARCHAR2(150)	Refer to base table.



---

## IPD\_FND\_OBJECTS\_EXT

IPD\_FND\_OBJECTS\_EXT stores objects associated with view privileges and SQL templates for object instance information. It is an extension to the FND\_OBJECTS table

### Column Descriptions

Name	Null?	Type	Description
OBJECT_NAME	NOT NULL	VARCHAR2(30)	Object name
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who column
VIEW_PRIVILEGE	NULL	VARCHAR2(30)	Privilege name, is alias to FUNCTION NAME
NAME_DESC_SELECT_SQL	NULL	VARCHAR2(1000)	SQL template to get object instance information
PARENT_OBJECT_ID	NULL	NUMBER	Parent Object Id. Refers to fnd_objects.object_id

---

## IPD\_FND\_OBJECTS\_V

IPD\_FND\_OBJECTS\_V is view of all fnd objects, and provides necessary attribute information for PDX.

### View Definition

```
CREATE VIEW IPD_FND_OBJECTS_V
as SELECT
    obj.object_id,
    obj.obj_name object_name,
    obj.application_id,
    obj.database_object_name,
    obj.pk1_column_name,
    obj.pk2_column_name,
    obj.pk3_column_name,
    obj.pk4_column_name,
    obj.pk5_column_name,
    obj_tl.display_name,
    obj_tl.description ,
    ext.view_privilege,
    ext.name_desc_select_sql,
    ext.parent_object_id
FROM
    fnd_objects obj,
    fnd_objects_tl obj_tl,
    ipd_fnd_objects_ext ext
Where obj.object_id= obj_tl.object_id
and   obj_tl.language= USERENV('LANG')
and   obj.obj_name = ext.object_name
```

### Column Descriptions

Name	Null?	Type	Description
OBJECT_ID	NOT NULL	NUMBER(15)	Refer to base table.
OBJECT_NAME	NOT NULL	VARCHAR2(30)	Refer to base table.
APPLICATION_ID	NOT NULL	NUMBER(15)	Refer to base table.
DATABASE_OBJECT_NAME	NOT NULL	VARCHAR2(30)	Refer to base table.
PK1_COLUMN_NAME	NOT NULL	VARCHAR2(30)	Refer to base table.
PK2_COLUMN_NAME	NULL	VARCHAR2(30)	Refer to base table.
PK3_COLUMN_NAME	NULL	VARCHAR2(30)	Refer to base table.
PK4_COLUMN_NAME	NULL	VARCHAR2(30)	Refer to base table.
PK5_COLUMN_NAME	NULL	VARCHAR2(30)	Refer to base table.
DISPLAY_NAME	NOT NULL	VARCHAR2(240)	Refer to base table.
DESCRIPTION	NOT NULL	VARCHAR2(2000)	Refer to base table.
VIEW_PRIVILEGE	NULL	VARCHAR2(30)	Refer to base table.
NAME_DESC_SELECT_SQL	NULL	VARCHAR2(1000)	Refer to base table.
PARENT_OBJECT_ID	NULL	NUMBER	Refer to base table.



## IPD\_ITEMS\_V

IPD\_ITEMS\_V is a view for all PDX items. SELLABLE\_FLAG indicates a purchasable item. If SECURE\_FLAG is Y, this is a development item and therefore displaying such an item is restricted by PDX security. OWNER\_PERSON\_ID identifies the owner of the item.

### View Definition

```

CREATE VIEW IPD_ITEMS_V
  as SELECT
    B.RT_ITEM_ID  ITEM_ID
  , B.SUPPLIER_ID PARTY_ID
  , B.A1 PARTY_NAME
  , B.A3 PARTY_ITEM_NUMBER
  , TL.A7 SHORT_DESCRIPTION
  , TL.A23 LONG_DESCRIPTION
  , B.A30 SELLABLE_FLAG
  , B.A31 SECURE_FLAG
  , B.A32 OWNER_PERSON_ID

  FROM
    ICX_POR_ITEMS_TL  TL
  , ICX_POR_ITEMS    B
WHERE
    B.RT_ITEM_ID = TL.RT_ITEM_ID
  AND TL.LANGUAGE = userenv('LANG')
  AND B.STATUS = 'PUBLISHED'

```

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
FND_ATTACHED_DOCUMENTS	PK1_VALUE	ITEM_ID
FND_GRANTS	OBJECT_ID	ITEM_ID

### Column Descriptions

Name	Null?	Type	Description
ITEM_ID	NOT NULL	NUMBER	Item unique identifier
PARTY_ID	NULL	NUMBER	Supplier Id
PARTY_NAME	NULL	VARCHAR2(255)	Supplier Name
PARTY_ITEM_NUMBER	NULL	VARCHAR2(255)	Supplier Item number
SHORT_DESCRIPTION	NULL	VARCHAR2(2000)	Short Description for item
LONG_DESCRIPTION	NULL	VARCHAR2(2000)	Long Description for item
SELLABLE_FLAG	NULL	VARCHAR2(1)	Purchasable flag
SECURE_FLAG	NULL	VARCHAR2(1)	Flag to indicate whether Development item
OWNER_PERSON_ID	NULL	NUMBER	Owner's Party Id

---

## IPD\_ITEM\_CROSS\_REFS\_ALL\_V

IPD\_ITEM\_CROSS\_REFS\_ALL\_V is the Item cross references view.

### View Definition

```
CREATE VIEW IPD_ITEM_CROSS_REFS_ALL_V
  as SELECT
      B.ROWID          ROW_ID
    , B.CROSS_REF_ID
    , B.ITEM_ID
    , B.REF_TYPE
    , B.REF_ITEM_ID
    , B.REF_IDENTIFIER
    , B.RECIPROCAL_FLAG
    , B.PREFERENCE_NUMBER
    , B.CREATION_DATE
    , B.CREATED_BY
    , B.LAST_UPDATE_DATE
    , B.LAST_UPDATED_BY
    , B.LAST_UPDATE_LOGIN
    , T.CROSS_REF_DESCRIPTION

  FROM
      IPD_ITEM_CROSS_REFS_TL T
    , IPD_ITEM_CROSS_REFS_B B
 WHERE
      B.CROSS_REF_ID = T.CROSS_REF_ID
    AND T.LANGUAGE = userenv('LANG')
UNION ALL
SELECT
      B.ROWID          ROW_ID
    , B.CROSS_REF_ID
    , B.REF_ITEM_ID   ITEM_ID
    , B.REF_TYPE
    , B.ITEM_ID      REF_ITEM_ID
    , B.REF_IDENTIFIER
    , B.RECIPROCAL_FLAG
    , B.PREFERENCE_NUMBER
    , B.CREATION_DATE
    , B.CREATED_BY
    , B.LAST_UPDATE_DATE
    , B.LAST_UPDATED_BY
    , B.LAST_UPDATE_LOGIN
    , T.CROSS_REF_DESCRIPTION
  FROM
      IPD_ITEM_CROSS_REFS_TL T
    , IPD_ITEM_CROSS_REFS_B B
 WHERE
      B.RECIPROCAL_FLAG = 'Y'
    AND B.CROSS_REF_ID = T.CROSS_REF_ID
    AND T.LANGUAGE = userenv('LANG')
```



Column Descriptions

<u>Name</u>	<u>Null?</u>	<u>Type</u>	<u>Description</u>
ROW_ID	NULL	ROWID	Refer to base table.
CROSS_REF_ID	NOT NULL	NUMBER	Refer to base table.
ITEM_ID	NOT NULL	NUMBER	Refer to base table.
REF_TYPE	NOT NULL	VARCHAR2(30)	Refer to base table.
REF_ITEM_ID	NOT NULL	NUMBER	Refer to base table.
REF_IDENTIFIER	NULL	VARCHAR2(240)	Refer to base table.
RECIPROCAL_FLAG	NOT NULL	VARCHAR2(1)	Refer to base table.
PREFERENCE_NUMBER	NULL	NUMBER	Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER	Refer to base table.
CROSS_REF_DESCRIPTION	NULL	VARCHAR2(4000)	Refer to base table.

## IPD\_ITEM\_CROSS\_REFS\_B

IPD\_ITEM\_CROSS\_REFS\_B stores the cross reference information for all items. Each row in this table identifies a unique cross reference for an item, and is identified by CROSS\_REF\_ID. Item cross references can be created between items within a company (or) with an item outside the company. The Item ID is stored in ITEM\_ID, and the cross referenced Item ID is stored in REF\_ITEM\_ID. The type of the cross reference can be identified by REF\_TYPE. A cross reference can be of the types: Identical / Related / Substitute. An item can be cross referenced to only one item of the outside company.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
ICX_POR_ITEMS	RT_ITEM_ID	ITEM_ID
ICX_POR_ITEMS	RT_ITEM_ID	REF_ITEM_ID

### Column Descriptions

Name	Null?	Type	Description
CROSS_REF_ID (PK)	NOT NULL	NUMBER	Cross Reference unique identifier
ITEM_ID	NOT NULL	NUMBER	Item Id
REF_TYPE	NOT NULL	VARCHAR2(30)	Cross Reference Type
REF_ITEM_ID	NOT NULL	NUMBER	Cross Referenced Item Id
REF_IDENTIFIER	NULL	VARCHAR2(240)	Cross Reference Identifier
RECIPROCAL_FLAG	NOT NULL	VARCHAR2(1)	Indicator whether Item Cross Reference is reciprocal
PREFERENCE_NUMBER	NULL	NUMBER	Preference Number. To indicate the priority for the Cross Reference. Not supported in PDX 6.1 release.
CREATION_DATE	NOT NULL	DATE	Standard Who Column
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column
REQUEST_ID	NULL	NUMBER	Standard Who Column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_CROSS_REFS_B_N1	NOT UNIQUE	1	ITEM_ID
		2	REF_TYPE
		3	REF_ITEM_ID
IPD_ITEM_CROSS_REFS_B_N2	NOT UNIQUE	1	REF_ITEM_ID
		2	REF_TYPE
		3	ITEM_ID
IPD_ITEM_CROSS_REFS_B_N3	NOT UNIQUE	1	REF_IDENTIFIER
		2	REF_TYPE
		3	ITEM_ID
IPD_ITEM_CROSS_REFS_B_U1	UNIQUE	1	CROSS_REF_ID

Oracle Proprietary, Confidential Information--Use Restricted by Contract



*Sequences*

<u>Sequence</u>	<u>Derived Column</u>
IPD_ITEM_CROSS_REFS_B_S	CROSS_REF_ID

---

## IPD\_ITEM\_CROSS\_REFS\_TL

IPD\_ITEM\_CROSS\_REFS\_TL stores translated information about the itemCross references in IPD\_ITEM\_CROSS\_REFS\_B. Each row includes the language the row is translated into, the translated attributes of the cross reference, and the cross reference identifier (CROSS\_REF\_ID) that uniquely identifies the item cross reference. You need one row for each cross reference in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_CROSS_REFS_B	CROSS_REF_ID	CROSS_REF_ID

### Column Descriptions

Name	Null?	Type	Description
CROSS_REF_ID (PK)	NOT NULL	NUMBER	Cross Reference unique identifier
LANGUAGE (PK)	NOT NULL	VARCHAR2(4)	Language
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source Language
CROSS_REF_DESCRIPTION	NULL	VARCHAR2(4000)	Cross Reference description
CREATION_DATE	NOT NULL	DATE	Standard Who Column
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_CROSS_REFS_TL_U1	UNIQUE	1 2	CROSS_REF_ID LANGUAGE



## IPD\_ITEM\_CROSS\_REFS\_VL

IPD\_ITEM\_CROSS\_REFS\_VL is the item cross references view by language.

### View Definition

```

CREATE VIEW IPD_ITEM_CROSS_REFS_VL
as SELECT
    B.ROWID                ROW_ID
,   B.CROSS_REF_ID
,   B.ITEM_ID
,   B.REF_TYPE
,   B.REF_ITEM_ID
,   B.REF_IDENTIFIER
,   B.RECIPROCAL_FLAG
,   B.PREFERENCE_NUMBER
,   B.CREATION_DATE
,   B.CREATED_BY
,   B.LAST_UPDATE_DATE
,   B.LAST_UPDATED_BY
,   B.LAST_UPDATE_LOGIN
,   T.CROSS_REF_DESCRIPTION

FROM
    IPD_ITEM_CROSS_REFS_TL T
,   IPD_ITEM_CROSS_REFS_B  B
WHERE
    B.CROSS_REF_ID = T.CROSS_REF_ID
AND T.LANGUAGE = userenv('LANG')

```

### Column Descriptions

Name	Null?	Type	Description
ROW_ID	NULL	ROWID	Refer to base table.
CROSS_REF_ID	NOT NULL	NUMBER	Refer to base table.
ITEM_ID	NOT NULL	NUMBER	Refer to base table.
REF_TYPE	NOT NULL	VARCHAR2(30)	Refer to base table.
REF_ITEM_ID	NOT NULL	NUMBER	Refer to base table.
REF_IDENTIFIER	NULL	VARCHAR2(240)	Refer to base table.
RECIPROCAL_FLAG	NOT NULL	VARCHAR2(1)	Refer to base table.
PREFERENCE_NUMBER	NULL	NUMBER	Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER	Refer to base table.
CROSS_REF_DESCRIPTION	NULL	VARCHAR2(4000)	Refer to base table.

## IPD\_ITEM\_PROJECTS\_B

IPD\_ITEM\_PROJECTS\_B stores the Association of Item/Revision to Project/Task. In case of Item Project Association, the revision ID is null. ASSOCIATION\_TYPE can be any of the following: DELIVERABLE, DESIGN, DEVELOPMENT, IMPLEMENTATION, TESTING.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_REVISIONS_B	REVISION_ID	REVISION_ID
PA_PROJECTS_ALL	PROJECT_ID	ITEM_PROJECT_ID
PA_TASKS	TASK_ID	TASK_ID

### Column Descriptions

Name	Null?	Type	Description
ITEM_PROJECT_ID (PK)	NOT NULL	NUMBER	Item / Revision to Project / Task Association unique Identifier
ITEM_ID	NOT NULL	NUMBER	Item identifier
PROJECT_ID	NOT NULL	NUMBER	Project identifier
TASK_ID	NULL	NUMBER	Task identifier
ASSOCIATION_TYPE	NOT NULL	VARCHAR2(30)	Association Type
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column
REVISION_ID	NULL	NUMBER	Item Revision identifier

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_PROJECTS_B_N1	NOT UNIQUE	1	PROJECT_ID
		2	ASSOCIATION_TYPE
		3	TASK_ID
		4	ITEM_ID
IPD_ITEM_PROJECTS_B_N2	NOT UNIQUE	1	ITEM_ID
		2	ASSOCIATION_TYPE
		3	PROJECT_ID
		4	TASK_ID
IPD_ITEM_PROJECTS_B_U1	UNIQUE	1	ITEM_PROJECT_ID

### Sequences

Sequence	Derived Column
IPD_ITEM_PROJECTS_B_S	ITEM_PROJECT_ID



## IPD\_ITEM\_PROJECTS\_TL

IPD\_ITEM\_PROJECTS\_TL stores translated information about the Item/Revision to Project/Task associations in IPD\_ITEM\_PROJECTS\_B. Each row includes the language the row is translated to, the translated attributes of the association, and the Association Identifier (ITEM\_PROJECT\_ID) that uniquely identifies the Item/Revision to Project/Task association. You need one row for each association in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_PROJECTS_B	ITEM_PROJECT_ID	ITEM_PROJECT_ID

### Column Descriptions

Name	Null?	Type	Description
ITEM_PROJECT_ID (PK)	NOT NULL	NUMBER	Item / Revision to Project / Task Association unique Identifier
LANGUAGE (PK)	NOT NULL	VARCHAR2(4)	Language
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source Language
ASSOCIATION_DESCRIPTION	NULL	VARCHAR2(4000)	Association Description
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_PROJECTS_TL_U1	UNIQUE	1	ITEM_PROJECT_ID
		2	LANGUAGE

## IPD\_ITEM\_PROJECTS\_VL

Item/Revision – Project/Task Association view.

### View Definition

```
CREATE VIEW IPD_ITEM_PROJECTS_VL
  as SELECT
      ip.ROWID          ROW_ID
    , ip.item_project_id
    , ip.item_id
    , ip.project_id
    , ip.task_id
    , ip.association_type
    , ip.created_by
    , ip.creation_date
    , ip.last_updated_by
    , ip.last_update_date
    , ip.last_update_login
    , ipl.association_description
    , ip.revision_id

  FROM
    ipd_item_projects_b ip
  , ipd_item_projects_tl ipl
 WHERE
    ip.item_project_id=ipl.item_project_id
 AND  ipl.language=userenv('LANG')
```

### Column Descriptions

Name	Null?	Type	Description
ROW_ID	NULL	ROWID	Refer to base table.
ITEM_PROJECT_ID	NOT NULL	NUMBER	Refer to base table.
ITEM_ID	NOT NULL	NUMBER	Refer to base table.
PROJECT_ID	NOT NULL	NUMBER	Refer to base table.
TASK_ID	NULL	NUMBER	Refer to base table.
ASSOCIATION_TYPE	NOT NULL	VARCHAR2(30)	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER	Refer to base table.
ASSOCIATION_DESCRIPTION	NULL	VARCHAR2(4000)	Refer to base table.
REVISION_ID	NOT NULL	NUMBER	Refer to base table.

## IPD\_ITEM\_REVISIONS\_B

IPD\_ITEM\_REVISIONS\_B stores the revision information for items. Each row in this table identifies a unique revision for an item, and is identified by REVISION\_ID. REVISION\_IDENTIFIER indicates the alphanumeric literal identifier for the revision, and can contain a maximum of 5 segments. Each segment can be a maximum of 10 characters in length. REVISION\_SORT\_SEQUENCE facilitates the sequencing of the revisions. REVISION\_SEGMENTS 1 to 5 store the broken down segment content from REVISION\_IDENTIFIER. REVISED\_FROM\_REVISION\_ID indicates the parent revision for the current revision.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
FND_ATTACHED_DOCUMENTS	PK1_VALUE	REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID	NOT NULL	NUMBER	Item Revision Unique Identifier
ITEM_ID	NOT NULL	NUMBER	Item unique identifier
REVISION_IDENTIFIER	NOT NULL	VARCHAR2(54)	Revision identifier
REVISION_SORT_SEQUENCE	NOT NULL	NUMBER	Revision sort sequence
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
PRIMARY_REASON_CODE	NOT NULL	VARCHAR2(30)	Primary Reason for Revision
START_DATE	NOT NULL	DATE	Revision Start Date
STATUS	NOT NULL	VARCHAR2(30)	Revision status
REVISION_SEGMENT1	NOT NULL	VARCHAR2(10)	Revision Segment1
REVISION_SEGMENT2	NULL	VARCHAR2(10)	Revision Segment2
REVISION_SEGMENT3	NULL	VARCHAR2(10)	Revision Segment3
REVISION_SEGMENT4	NULL	VARCHAR2(10)	Revision Segment4
REVISION_SEGMENT5	NULL	VARCHAR2(10)	Revision Segment5
REVISED_FROM_REVISION_ID	NULL	NUMBER	Revised from Revision Identifier
END_DATE	NULL	DATE	End date for the Revision
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column
REQUEST_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_REVISIONS_B_N1	NOT UNIQUE	5	ITEM_ID
IPD_ITEM_REVISIONS_B_U1	UNIQUE	5	ITEM_ID
		10	REVISION_IDENTIFIER
IPD_ITEM_REVISIONS_B_U2	NOT UNIQUE	5	REVISION_ID
IPD_ITEM_REVISIONS_B_U3	NOT UNIQUE	5	ITEM_ID
		10	REVISION_SORT_SEQUENCE



*Sequences*

<u>Sequence</u>	<u>Derived Column</u>
IPD_ITEM_REVISIONS_B_S	REVISION_ID



---

## IPD\_ITEM\_REVISIONS\_TL

IPD\_ITEM\_REVISIONS\_TL stores translated information about the item revisions in IPD\_ITEM\_REVISIONS\_B. Each row includes the language the row is translated into, the translated attributes of the revision, and the revision identifier (REVISION\_ID) that uniquely identifies the item cross reference. You need one row for each revision in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_REVISIONS_B	REVISION_ID	REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID	NOT NULL	NUMBER	Item Revision Unique Identifier
LANGUAGE	NOT NULL	VARCHAR2(4)	Language
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source Language
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
SHORT_DESCRIPTION	NOT NULL	VARCHAR2(240)	Short Description
LONG_DESCRIPTION	NULL	VARCHAR2(4000)	Long Description
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_ITEM_REVISIONS_TL_U1	NOT UNIQUE	5 10	REVISION_ID LANGUAGE

## IPD\_ITEM\_REVISIONS\_VL

Item/Revision view by language.

### View Definition

```
CREATE VIEW IPD_ITEM_REVISIONS_VL
as SELECT
    B.REVISION_ID,
    B.ITEM_ID,
    B.REVISION_IDENTIFIER,
    B.REVISION_SORT_SEQUENCE,
    B.CREATED_BY,
    B.CREATION_DATE,
    B.LAST_UPDATED_BY,
    B.LAST_UPDATE_DATE,
    B.PRIMARY_REASON_CODE,
    B.START_DATE,
    B.STATUS,
    B.REVISION_SEGMENT1,
    B.REVISION_SEGMENT2,
    B.REVISION_SEGMENT3,
    B.REVISION_SEGMENT4,
    B.REVISION_SEGMENT5 ,
    B.REVISED_FROM_REVISION_ID,
    B.END_DATE,
    B.LAST_UPDATE_LOGIN,
    B.REQUEST_ID ,
    B.PROGRAM_APPLICATION_ID,
    B.PROGRAM_ID,
    B.PROGRAM_UPDATE_DATE,
    TL.SHORT_DESCRIPTION,
    TL.LONG_DESCRIPTION
FROM IPD_ITEM_REVISIONS_B B, IPD_ITEM_REVISIONS_TL TL
Where B.REVISION_ID = TL.REVISION_ID
AND    TL.LANGUAGE = USERENV('LANG')
```

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID	NOT NULL	NUMBER	Refer to base table.
ITEM_ID	NOT NULL	NUMBER	Refer to base table.
REVISION_IDENTIFIER	NOT NULL	VARCHAR2(54)	Refer to base table.
REVISION_SORT_SEQUENCE	NOT NULL	NUMBER	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
PRIMARY_REASON_CODE	NOT NULL	VARCHAR2(30)	Refer to base table.
START_DATE	NOT NULL	DATE	Refer to base table.
STATUS		NULL VARCHAR2(30)	Refer to base table.
REVISION_SEGMENT1	NOT NULL	VARCHAR2(10)	Refer to base table.
REVISION_SEGMENT2		NULL VARCHAR2(10)	Refer to base table.
REVISION_SEGMENT3		NULL VARCHAR2(10)	Refer to base table.
REVISION_SEGMENT4		NULL VARCHAR2(10)	Refer to base table.

Oracle Proprietary, Confidential Information--Use Restricted by Contract

REVISION_SEGMENT5	NULL	VARCHAR2(10)	Refer to base table.
REVISED_FROM_REVISION_ID	NULL	NUMBER	Refer to base table.
END_DATE	NULL	DATE	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER	Refer to base table.
REQUEST_ID	NULL	NUMBER	Refer to base table.
PROGRAM_APPLICATION_ID	NULL	NUMBER	Refer to base table.
PROGRAM_ID	NULL	NUMBER	Refer to base table.
PROGRAM_UPDATE_DATE	NULL	DATE	Refer to base table.
SHORT_DESCRIPTION	NOT NULL	VARCHAR2(240)	Refer to base table.
LONG_DESCRIPTION	NULL	VARCHAR2(4000)	Refer to base table.

---

## IPD\_OBJECT\_ROLES

This table is used to maintain role status, and identifies whether or not a role is enabled or disabled.

### Column Descriptions

Name	Null?	Type	Description
OBJECT_ID	NOT NULL	NUMBER	Object ID. Reference to fnd_objects object_id
ROLE_ID	NOT NULL	NUMBER	Reference to fnd_menus.menu_id
ENABLED_FLAG	NOT NULL	VARCHAR2(1)	Flag to enable or disable a role
CREATED_BY	NOT NULL	NUMBER	Standard who column
CREATION_DATE	NOT NULL	DATE	Standard who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard who column
LAST_UPDATE_LOGIN	NULL	NUMBER	Standard who column

---

## IPD\_PRODUCT\_DEFECTS

IPD\_PRODUCT\_DEFECTS table contains information about product defects.

### Column Descriptions

Name	Null?	Type	Description
DEFECT_ID (PK)	NOT NULL	NUMBER	Defect identifier
DEFECT_NUMBER	NULL	VARCHAR2(30)	Defect number
SEVERITY_CODE	NULL	VARCHAR2(30)	Severity code
SEVERITY_DESC	NULL	VARCHAR2(80)	Severity description
PRIORITY_CODE	NULL	VARCHAR2(30)	Priority code
PRIORITY_DESC	NULL	VARCHAR2(80)	Priority description
STATUS_CODE	NULL	VARCHAR2(30)	Status code
STATUS_DESC	NULL	VARCHAR2(80)	Status description
ASSIGNED_TO_ID	NULL	NUMBER	Assigned to person identifier
CUSTOMER_ID	NULL	NUMBER	Customer identifier
PRODUCT_ID	NULL	NUMBER	Product identifier
PRODUCT_REVISION_ID	NULL	NUMBER	Product revision identifier
PRODUCT_COMPONENT_ID	NULL	NUMBER	Product component identifier
SHORT_DESCRIPTION	NULL	VARCHAR2(240)	Short description
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
CREATED_BY	NOT NULL	NUMBER	Standard Who column

## IPD\_PRODUCT\_DEFECTS\_V

IPD\_PRODUCT\_DEFECTS\_V describes the product defect information, and should be customized according to the customer's bug reporting system when implemented.

### View Definition

```
CREATE VIEW IPD_PRODUCT_DEFECTS_V
  as SELECT
    defect_id,
    defect_number,
    severity_code,
    severity_desc,
    priority_code,
    priority_desc,
    status_code,
    status_desc,
    assigned_to_id,
    null,
    customer_id,
    null,
    product_id,
    null,
    product_revision_id,
    null,
    product_component_id,
    null,
    short_description,
    null,
    last_update_date,
    last_updated_by,
    null,
    creation_date,
    created_by,
    null
  from ipd_product_defects
```

### Column Descriptions

Name	Null?	Type	Description
DEFECT_ID	NOT NULL	NUMBER	Refer to base table.
DEFECT_NUMBER	NULL	VARCHAR2(30)	Refer to base table.
SEVERITY_CODE	NULL	VARCHAR2(30)	Refer to base table.
SEVERITY_DESC	NULL	VARCHAR2(80)	Refer to base table.
PRIORITY_CODE	NULL	VARCHAR2(30)	Refer to base table.
PRIORITY_DESC	NULL	VARCHAR2(80)	Refer to base table.
STATUS_CODE	NULL	VARCHAR2(30)	Refer to base table.
STATUS_DESC	NULL	VARCHAR2(80)	Refer to base table.
ASSIGNED_TO_ID	NULL	NUMBER	Refer to base table.
ASSIGNED_TO_PERSON	NULL		Refer to base table.
CUSTOMER_ID	NULL	NUMBER	Refer to base table.
CUSTOMER_NAME	NULL		Refer to base table.
PRODUCT_ID	NULL	NUMBER	Refer to base table.
PRODUCT_NAME	NULL		Refer to base table.

Oracle Proprietary, Confidential Information--Use Restricted by Contract

PRODUCT_REVISION_ID	NULL	NUMBER	Refer to base table.
PRODUCT_REVISION_NAME	NULL		Refer to base table.
PRODUCT_COMPONENT_ID	NULL	NUMBER	Refer to base table.
PRODUCT_COMPONENT_NAME	NULL		Refer to base table.
SHORT_DESCRIPTION	NULL	VARCHAR2(240)	Refer to base table.
URL	NULL		Refer to base table.
LAST_UPDATE_DATE	NOT NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NOT NULL	NUMBER	Refer to base table.
LAST_UPDATED_BY_PERSON	NULL		Refer to base table.
CREATION_DATE	NOT NULL	DATE	Refer to base table.
CREATED_BY	NOT NULL	NUMBER	Refer to base table.
CREATED_BY_PERSON	NULL		Refer to base table.

---

## IPD\_PRODUCT\_STRUCTURES

IPD\_PRODUCT\_STRUCTURES stores the header information for all product structures. Each row in this table identifies a unique product structure and is identified by PRODUCT\_STRUCTURE\_ID. If an item has more than one product structure, then the product structures are distinguished by NAME. A product structure can be either Engineering, Manufacturing or Marketing. The type of product structure can be identified by TYPE\_ID.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_PS_TYPES	TYPE_ID	TYPE_ID

### Column Descriptions

Name	Null?	Type	Description
PRODUCT_STRUCTURE_ID (PK)	NOT NULL	NUMBER	Product Structure unique identifier
PARENT_ITEM_ID	NOT NULL	NUMBER	Parent item identifier
NAME	NULL	VARCHAR2(30)	Product Structure name
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
TYPE_ID	NOT NULL	NUMBER	Product Structure type identifier
BATCH_SIZE	NULL	NUMBER	Batch size
DESCRIPTION	NULL	VARCHAR2(2000)	Product Structure description
REQUEST_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_ID	NULL	NUMBER	Concurrent Who Column
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PRODUCT_STRUCTURES_U1	UNIQUE	1	PRODUCT_STRUCTURE_ID
IPD_PRODUCT_STRUCTURES_U2	UNIQUE	1 2	PARENT_ITEM_ID NAME



---

## IPD\_PRODUCT\_STRUCTURES\_TL

IPD\_PRODUCT\_STRUCTURES\_TL stores translated information about the product structures in IPD\_PRODUCT\_STRUCTURES. Each row includes the language the row is translated into, the translated attributes of the product structure, and the product structure identifier, which uniquely identifies the product structure. You need one row for each product structure in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Column Descriptions

Name	Null?	Type	Description
PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product Structure identifier
LANGUAGE	NOT NULL	VARCHAR2(4)	Language code
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source language
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
DESCRIPTION	NULL	VARCHAR2(2000)	Product Structure description

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PRODUCT_STRUCTURES_TL_U1	UNIQUE	1	PRODUCT_STRUCTURE_ID
		2	LANGUAGE

## IPD\_PS\_COMPONENTS

IPD\_PS\_COMPONENTS stores information about product structure components. This table only stores single-level components. COMPONENT\_ID uniquely identifies a component row in this table. The parent is identified using PRODUCT\_STRUCTURE\_ID. A component can be Date effective or Revision effective. The effectivity type of the component is identified by EFFECTIVITY\_TYPE. EFFECTIVITY\_DATE and DISABLE\_DATE are used to time phase components into a product structure. EFFECTIVE\_FROM and EFFECTIVE\_TO allow a component to a revision effective as opposed to date effective. Same component can appear on a product structure multiple time provided the component effectivities do not overlap. COMPONENT\_SEQ\_NUM is unique within a product structure.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_REVISIONS_B	REVISION_ID ITEM_ID	ITEM_REVISION_ID COMPONENT_ITEM_ID
IPD_PRODUCT_STRUCTURES	PRODUCT_STRUCTURE_ID	PRODUCT_STRUCTURE_ID

### Column Descriptions

Name	Null?	Type	Description
COMPONENT_ID (PK)	NOT NULL	NUMBER	Component unique identifier
PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product Structure unique identifier
COMPONENT_ITEM_ID	NOT NULL	NUMBER	Component item identifier
EFFECTIVITY_TYPE	NULL	VARCHAR2(30)	Effectivity type identifier
EFFECTIVITY_DATE	NOT NULL	DATE	Start effective date of the component
DISABLE_DATE	NULL	DATE	Disable date of the component
EFFECTIVE_FROM	NULL	VARCHAR2(80)	Effective from. Used as a alternate mechanism to date effectivity
EFFECTIVE_TO	NULL	VARCHAR2(80)	Effective to. Used as a alternate mechanism to date effectivity
COMPONENT_QUANTITY	NOT NULL	NUMBER	Quantity of component on the Product Structure
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
CHANGE_NOTICE	NULL	VARCHAR2(30)	Engineering Change Order identifier
COMPONENT_SEQ_NUM	NULL	NUMBER	Component sequence within the Product Structure.
REQUEST_ID	NULL	NUMBER	Concurrent Who Columns
PROGRAM_APPLICATION_ID	NULL	NUMBER	Concurrent Who Columns
PROGRAM_ID	NULL	NUMBER	Concurrent Who Columns
PROGRAM_UPDATE_DATE	NULL	DATE	Concurrent Who Columns
ITEM_REVISION_ID	NULL	NUMBER	Item revision identifier

Oracle Proprietary, Confidential Information--Use Restricted by Contract



*Indexes*

<u>Index Name</u>	<u>Index Type</u>	<u>Sequence</u>	<u>Column Name</u>
IPD_PS_COMPONENTS_U1	UNIQUE	1	COMPONENT_ID
IPD_PS_COMPONENTS_U2	UNIQUE	1	PRODUCT_STRUCTURE_ID
		2	COMPONENT_ITEM_ID
		3	EFFECTIVITY_DATE
		4	EFFECTIVITY_TYPE
		5	EFFECTIVE_FROM

## IPD\_PS\_COMPONENTS\_INTERFACE

IPD\_PS\_COMPONENTS\_INTERFACE temporarily stores information about product structure components. This table is used by the product structure bulkload to temporarily insert component information and then call the Product Structure Business Object. The Business Object is analogous to an open interface. It picks the rows, processes them and once finished the bulk load process will clean the rows that it inserted. The table contains user-friendly columns that allow you to easily enter data. The Business Object will derive any values based on the user-friendly columns. For example, if you enter COMPONENT\_ITEM\_NAME, the business object will derive the COMPONENT\_ITEM\_ID.

### Column Descriptions

Name	Null?	Type	Description
COMPONENT_ITEM_NAME	NULL	VARCHAR2(240)	Name of the component item
COMPONENT_ITEM_REVISION	NULL	VARCHAR2(54)	Item Revision this component is tied with
PARENT_ITEM_NAME	NULL	VARCHAR2(240)	Name of the parent item
PARENT_SUPPLIER_NAME	NULL	VARCHAR2(80)	Supplier of the Parent item
COMPONENT_SUPPLIER_NAME	NULL	VARCHAR2(80)	Name of the supplier for the component item
PS_NAME	NULL	VARCHAR2(30)	Product Structure Name
EFFECTIVITY_TYPE	NULL	VARCHAR2(30)	Effectivity type of this component
START_EFFECTIVE_DATE	NULL	DATE	Date from which this component is effective
DISABLE_DATE	NULL	DATE	Date when this component is disabled
EFFECTIVE_FROM	NULL	VARCHAR2(80)	Effective from. Used as a alternate mechanism to date effectivity
EFFECTIVE_TO	NULL	VARCHAR2(80)	
COMPONENT_QUANTITY	NULL	NUMBER	Quantity of component on the Product Structure
CHANGE_NOTICE	NULL	VARCHAR2(30)	Engineering change order identifier
COMPONENT_SEQ_NUM	NULL	NUMBER	Sequence of the component within the product structure
TRANSACTION_TYPE	NULL	VARCHAR2(15)	Transaction type. Can be Create, Update or Delete. Sync when bulkload uses it.
RETURN_STATUS	NULL	VARCHAR2(1)	Return status assigned to this component after processing, whether success or failure
TRANSACTION_ID	NULL	NUMBER	Transaction Identifier. Many transactions can be grouped into a transaction group
TRANSACTION_GROUP_ID	NULL	NUMBER	

## IPD\_PS\_EXPLOSIONS

IPD\_PS\_EXPLOSIONS stores permanent explosions for product structures.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_PRODUCT_STRUCTURES	PRODUCT_STRUCTURE_ID	COMP_PRODUCT_STRUCTURE_ID
IPD_PRODUCT_STRUCTURES	PRODUCT_STRUCTURE_ID	PRODUCT_STRUCTURE_ID
IPD_PRODUCT_STRUCTURES	PRODUCT_STRUCTURE_ID	TOP_PRODUCT_STRUCTURE_ID
IPD_PS_COMPONENTS	COMPONENT_ID	COMPONENT_ID

### Column Descriptions

Name	Null?	Type	Description
TOP_PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product Structure identifier for level 0 product structures
TOP_ITEM_ID	NOT NULL	NUMBER	Item identifier of level 0 items
PS_NAME	NULL	VARCHAR2(30)	Product structure name
PS_LEVEL	NOT NULL	NUMBER	Level in the Product Structure
COMPONENT_ID	NOT NULL	NUMBER	Component unique identifier
PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product Structure unique identifier
COMPONENT_ITEM_ID	NOT NULL	NUMBER	Component item identifier
PARENT_ITEM_ID	NOT NULL	NUMBER	Parent item identifier
EFFECTIVITY_TYPE	NULL	VARCHAR2(30)	Indicates the Effectivity type of the component
EFFECTIVITY_DATE	NOT NULL	DATE	Start effective date of the component
DISABLE_DATE	NULL	DATE	Disable date of the component
EFFECTIVE_FROM	NULL	VARCHAR2(80)	Effective From. Used for alternative effectivity.
EFFECTIVE_TO	NULL	VARCHAR2(80)	Effective To. Used in conjunction with Effective From
COMPONENT_QUANTITY	NOT NULL	NUMBER	Quantity of component on the Product Structure
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
ITEM_REVISION	NULL	VARCHAR2(54)	Revision of the item
CHANGE_NOTICE	NULL	VARCHAR2(30)	Engineering Change order identifier
COMPONENT_SEQ_NUM	NULL	NUMBER	Sequence of the component within the product structure
REEXPLODE_FLAG	NULL	NUMBER	Re-Explosion flag. Indicates if the product structure has been modified since the last explosion and if a re-explosion is required.
SORT_ORDER	NULL	VARCHAR2(240)	Sorting sequence for the explosion
COMPONENT_CODE	NULL	VARCHAR2(2000)	Cconcatenation of Component item Ids
LOOP_FLAG	NULL	NUMBER	Indicates if a loop is detected
COMP_PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Component's Product Structure Id



*Indexes*

<u>Index Name</u>	<u>Index Type</u>	<u>Sequence</u>	<u>Column Name</u>
IPD_PS_EXPLOSIONS_N1	NOT UNIQUE	10	TOP_PRODUCT_STRUCTURE_ID



---

## IPD\_PS\_HEADER\_INTERFACE

IPD\_PS\_HEADER\_INTERFACE temporarily stores information about the product structure header. This table is used by the product structure bulkload to temporarily insert header information and then call the Product Structure Business Object. The Business Object is analogous to an open interface. It picks the rows, processes them and once finished the bulk load process cleans the rows that it inserted. The table contains user-friendly columns that allow you to easily enter data. The Business Object will derive any values based on the user-friendly columns. For example, if you enter PARENT\_ITEM\_NAME, the business object will derive the COMPONENT\_ITEM\_ID.

### Column Descriptions

Name	Null?	Type	Description
PARENT_ITEM_NAME	NULL	VARCHAR2(240)	Name of the parent item
PS_NAME	NULL	VARCHAR2(30)	Product Structure name
SUPPLIER_NAME	NULL	VARCHAR2(80)	Name of the supplier who supplies the parent item
TYPE_ID	NULL	NUMBER	Product Structure type identifier
BATCH_SIZE	NULL	NUMBER	Batch size
DESCRIPTION	NULL	VARCHAR2(2000)	Product Structure description
TRANSACTION_TYPE	NULL	VARCHAR2(15)	Transaction type. Supports CREATE, UPDATE, DELETE and SYNC when used by BulkLoad
RETURN_STATUS	NULL	VARCHAR2(1)	Return status for this record
TRANSACTION_ID	NULL	NUMBER	Transaction identifier within the transaction group
TRANSACTION_GROUP_ID	NULL	NUMBER	Transaction Group identifier

---

## IPD\_PS\_PROCESS\_ASSIGNMENTS

IPD\_PS\_PROCESS\_ASSIGNMENTS table is currently not used.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_REVISIONS_B	ITEM_ID REVISION_ID	ITEM_ID ITEM_REVISION_ID
IPD_PRODUCT_STRUCTURES	NAME PARENT_ITEM_ID	PRODUCT_STRUCTURE_NAME ITEM_ID

### Column Descriptions

Name	Null?	Type	Description
ASSIGNMENT_ID	NOT NULL	NUMBER	Assignment Identifier
ITEM_ID	NOT NULL	NUMBER	Item Identifier
ITEM_REVISION_ID	NOT NULL	NUMBER	Item Revision Identifier
PRODUCT_STRUCTURE_NAME	NOT NULL	VARCHAR2(10)	Product Structure Name
PROCESS_NAME	NOT NULL	VARCHAR2(10)	Process Name
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PS_PROCESS_ASSIGNMENTS_U1	UNIQUE	1	ASSIGNMENT_ID
IPD_PS_PROCESS_ASSIGNMENTS_U2	UNIQUE	1	ITEM_ID
		2	ITEM_REVISION_ID
		3	PRODUCT_STRUCTURE_NAME
		4	PROCESS_NAME

## IPD\_PS\_REVISIONS

IPD\_PS\_REVISIONS store the revisions for a product structure. REVISION\_ID identifiers are unique revision records. The product structure to which the revisions belong is identified by PRODUCT\_STRUCTURE\_ID. Alternatively, REVISION and PRODUCT\_STRUCTURE\_ID also identify a unique record. A product structure can have more than one revision, but only one revision can be effective at any given time.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_ITEM_REVISIONS_B	REVISION_IDENTIFIER	ITEM_REVISION_ID
IPD_PRODUCT_STRUCTURES	PRODUCT_STRUCTURE_ID	PRODUCT_STRUCTURE_ID
IPD_PS_REVISIONS	REQUEST_ID	REVISED_FROM_REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID	NOT NULL	NUMBER	Revision unique identifier
PRODUCT_STRUCTURE_ID	NOT NULL	NUMBER	Product structure identifier
REVISION	NOT NULL	VARCHAR2(54)	Revision name
CREATED_BY	NOT NULL	NUMBER	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
EFFECTIVITY_DATE	NOT NULL	DATE	Start effective date
DISABLE_DATE		DATE	Disable date
REVISION_SEGMENT1	NOT NULL	VARCHAR2(10)	Revision segment1
REVISION_SEGMENT2		VARCHAR2(10)	Revision segment1
REVISION_SEGMENT3		VARCHAR2(10)	Revision segment3
REVISION_SEGMENT4		VARCHAR2(10)	Revision segment4
REVISION_SEGMENT5		VARCHAR2(10)	Revision segment5
REVISION_SORT_SEQ_NUM	NOT NULL	NUMBER	Sorting sequence for this revision
REVISED_FROM_REVISION_ID		NUMBER	Previous revision identifier
DESCRIPTION		VARCHAR2(2000)	Revision description
ITEM_REVISION_ID		NUMBER	Item revision with which this product structure revision is associated
STATUS		VARCHAR2(30)	Revision status
PRIMARY_REASON_CODE		VARCHAR2(30)	Reason code identifier. Identifier the reason this revision was created.
CHANGE_NOTICE		VARCHAR2(30)	Engineering change order identifier
REQUEST_ID		NUMBER	Concurrent Who column
PROGRAM_APPLICATION_ID		NUMBER	Concurrent Who column
PROGRAM_ID		NUMBER	Concurrent Who column
PROGRAM_UPDATE_DATE		DATE	Concurrent Who column

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PS_REVISIONS_U1	UNIQUE	1	REVISION_ID



IPD_PS_REVISIONS_U2	UNIQUE	1	REVISION
		2	PRODUCT_STRUCTURE_ID



---

## IPD\_PS\_REVISIONS\_INTERFACE

IPD\_PS\_REVISIONS\_INTERFACE temporarily stores revision information for a product structure. This information is used by the Product Structure BulkLoad.

### Column Descriptions

Name	Null?	Type	Description
REVISION	NULL	VARCHAR2(54)	Revision name
PARENT_ITEM_NAME	NULL	VARCHAR2(240)	Parent item name
PS_NAME	NULL	VARCHAR2(30)	Product Structure Name
SUPPLIER_NAME	NULL	VARCHAR2(240)	Name of the supplier, supplying the parent item
PARENT_ITEM_REVISION	NULL	VARCHAR2(54)	Revision of the parent item
STATUS	NULL	VARCHAR2(30)	Revision status
START_EFFECTIVE_DATE	NULL	DATE	Start effectivity date of this Revision
DISABLE_DATE	NULL	DATE	Disable date of this Revision
DESCRIPTION	NULL	VARCHAR2(2000)	Revision description
PRIMARY_REASON_CODE	NULL	VARCHAR2(30)	Primary reason code for creating this revision
TRANSACTION_TYPE	NULL	VARCHAR2(10)	Transaction type. Supports CREATE, UPDATE, DELETE.
RETURN_STATUS	NULL	VARCHAR2(1)	Return status of this revision record
TRANSACTION_ID	NULL	NUMBER	Transaction identifier within the transaction group
TRANSACTION_GROUP_ID	NULL	NUMBER	Transaction Group identifier

---

## IPD\_PS\_REVISIONS\_TL

IPD\_PS\_REVISIONS\_TL stores translated information about the product structures revisions in IPD\_PS\_REVISIONS. Each row includes the language the row is translated into, the translated attributes of the revision, and the revision identifier that uniquely identifies the revision. You need one row for each product structure revision in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
IPD_CHANGE_REVISIONS	REVISION_ID	REVISION_ID

### Column Descriptions

Name	Null?	Type	Description
REVISION_ID	NOT NULL	NUMBER	Revision unique identifier
LANGUAGE	NOT NULL	VARCHAR2(4)	Language identifier
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source language identifier
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
DESCRIPTION	NULL	VARCHAR2(2000)	Revision description in the language identified by LANGUAGE

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PS_REVISIONS_TL_U1	UNIQUE	1	REVISION_ID
		2	LANGUAGE



---

## IPD\_PS\_TYPES

IPD\_PS\_TYPES stores the various types of product structures that can be created. TYPE\_ID uniquely identifies a record. There are 3 types that are seeded: Engineering, Manufacturing, and Marketing

### Column Descriptions

Name	Null?	Type	Description
TYPE_ID (PK)	NOT NULL	NUMBER	Product Structure type unique identifier
NAME	NOT NULL	VARCHAR2(30)	Product Structure Type name.
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
DESCRIPTION	NULL	VARCHAR2(2000)	Product Structure Type description

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PS_TYPES_U1	UNIQUE	1	TYPE_ID
IPD_PS_TYPES_U2	UNIQUE	1	NAME

---

## IPD\_PS\_TYPES\_TL

IPD\_PS\_TYPES\_TL stores translated information about the product structure types in IPD\_PS\_TYPES. Each row includes the language the row is translated into, the translated attributes of the type, and the type identifier that uniquely identifies the record. You need one row for each type in each of the installed languages. Each row includes a language and SOURCE\_LANG column that can be used to determine whether or not a value has been translated or defaulted.

### Column Descriptions

Name	Null?	Type	Description
TYPE_ID	NOT NULL	NUMBER	Type description
LANGUAGE	NOT NULL	VARCHAR2(4)	Language identifier
SOURCE_LANG	NOT NULL	VARCHAR2(4)	Source language identifier
CREATED_BY	NOT NULL	NUMBER	Standard Who column
CREATION_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER	Standard Who column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
DESCRIPTION	NULL	VARCHAR2(2000)	Type description translated in the language identified by LANGUAGE

### Indexes

Index Name	Index Type	Sequence	Column Name
IPD_PS_TYPES_TL_U1	UNIQUE	1	TYPE_ID
		2	LANGUAGE

## PA\_PROJECTS\_ALL

PA\_PROJECTS\_ALL stores the highest units of work defined in Oracle Projects at which funding can be allocated and at which revenue, invoices, unbilled receivables, and unearned revenue are generated and maintained. At the project level, users can specify the methods to be used for revenue accrual and billing, and the invoice formats to be used for invoicing. A project is broken down into one or more tasks. All projects can be funded by one or more agreements.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
HR_ALL_ORGANIZATION_UNITS	ORGANIZATION_ID	CARRYING_OUT_ORGANIZATION_ID
PA_BILLING_CYCLES	BILLING_CYCLE_ID	BILLING_CYCLE_ID
PA_CC_TP_SCHEDULES_BG	TP_SCHEDULE_ID	LABOR_TP_SCHEDULE_ID
PA_CC_TP_SCHEDULES_BG	TP_SCHEDULE_ID	NL_TP_SCHEDULE_ID
PA_DISTRIBUTION_RULES	DISTRIBUTION_RULE	DISTRIBUTION_RULE
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	COST_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	OVR_COST_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	REV_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	OVR_REV_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	INV_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	OVR_INV_IND_RATE_SCH_ID
PA_INVOICE_FORMATS	INVOICE_FORMAT_ID	LABOR_INVOICE_FORMAT_ID
PA_INVOICE_FORMATS	INVOICE_FORMAT_ID	NON_LABOR_INVOICE_FORMAT_ID
PA_INVOICE_FORMATS	INVOICE_FORMAT_ID	RETENTION_INVOICE_FORMAT_ID
PA_PROJECT_STATUSES	PROJECT_STATUS_CODE	PROJECT_STATUS_CODE
PA_PROJECT_TYPES_ALL	PROJECT_TYPE	PROJECT_TYPE
	ORG_ID	ORG_ID
PA_ROLE_LISTS	ROLE_LIST_ID	ROLE_LIST_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	ORGANIZATION_ID	LABOR_BILL_RATE_ORG_ID
	STD_BILL_RATE_SCHEDULE	LABOR_STD_BILL_RATE_SCHDL
	ORG_ID	ORG_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	ORGANIZATION_ID	NON_LABOR_BILL_RATE_ORG_ID
	STD_BILL_RATE_SCHEDULE	NON_LABOR_STD_BILL_RATE_SCHDL
	ORG_ID	ORG_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	BILL_RATE_SCH_ID	JOB_BILL_RATE_SCHEDULE_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	BILL_RATE_SCH_ID	EMP_BILL_RATE_SCHEDULE_ID
PER_JOB_GROUPS	JOB_GROUP_ID	BILL_JOB_GROUP_ID
PER_JOB_GROUPS	JOB_GROUP_ID	COST_JOB_GROUP_ID

### QuickCodes Columns

Column	QuickCodes Type	QuickCodes Table
LABOR_SCH_TYPE	PROJECT SCHEDULE TYPE	PA_LOOKUPS
	B	Bill Rate
	I	Indirect
NON_LABOR_SCH_TYPE	PROJECT SCHEDULE TYPE	PA_LOOKUPS
	B	Bill Rate
	I	Indirect
PM_PRODUCT_CODE	PM_PRODUCT_CODE	PA_LOOKUPS
PROJECT_STATUS_CODE	PROJECT STATUS	PA_LOOKUPS
	ACTIVE	Active
	CLOSED	Closed

Oracle Proprietary, Confidential Information--Use Restricted by Contract

WF_STATUS_CODE	WF_STATUS_CODE	PA_LOOKUPS
	IN_ROUTE	In Progress
	REJECTED	Rejected

Column Descriptions

Name	Null?	Type	Description
PROJECT_ID (PK)	NOT NULL	NUMBER(15)	The system-generated number that uniquely identifies the project
NAME	NOT NULL	VARCHAR2(30)	User-defined name that uniquely identifies the project
SEGMENT1	NOT NULL	VARCHAR2(25)	The project number that uniquely identifies the project. The project number can be user-entered or system-generated as defined in the implementation options
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER(15)	Standard Who Column
CREATION_DATE	NOT NULL	DATE	Standard Who Column
CREATED_BY	NOT NULL	NUMBER(15)	Standard Who Column
LAST_UPDATE_LOGIN	NOT NULL	NUMBER(15)	Standard Who Column
PROJECT_TYPE	NOT NULL	VARCHAR2(20)	The project type that classifies the project and defaults project information upon project entry
CARRYING_OUT_ORGANIZATION_ID	NOT NULL	NUMBER(15)	The identifier of the organization that is responsible for the project work
PUBLIC_SECTOR_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates whether the project is in the public sector or the private sector
PROJECT_STATUS_CODE	NOT NULL	VARCHAR2(30)	The status of the project.
DESCRIPTION	NULL	VARCHAR2(250)	The description of the project
START_DATE	NULL	DATE	The date on which the project starts; expenditure items with item dates before the start date cannot be entered for the project
COMPLETION_DATE	NULL	DATE	The date on which the project is completed; expenditure items with item dates after the completion date cannot be entered for the project
CLOSED_DATE	NULL	DATE	The date that the project was closed by changing the project status to Closed
DISTRIBUTION_RULE	NULL	VARCHAR2(30)	The distribution rule that specifies the contract project's revenue accrual and billing method
LABOR_INVOICE_FORMAT_ID	NULL	NUMBER(15)	The identifier of the invoice format used to group and format labor expenditure items into invoice lines
NON_LABOR_INVOICE_FORMAT_ID	NULL	NUMBER(15)	The identifier of the invoice format used to group and format non-labor expenditure items into invoice lines
RETENTION_INVOICE_FORMAT_ID	NULL	NUMBER(15)	The identifier of the invoice format used to format a retention line on the project's invoices

Oracle Proprietary, Confidential Information--Use Restricted by Contract

RETENTION_PERCENTAGE	NULL	NUMBER(15)	The percentage to be retained on each invoice generated for the project
BILLING_OFFSET	NULL	NUMBER(15)	The number of days after the project start date when the first invoice for the project is created
BILLING_CYCLE	NULL	NUMBER(15)	Obsolete
LABOR_STD_BILL_RATE_SCHDL	NULL	VARCHAR2(20)	The labor standard bill rate schedule that defaults to new tasks created for the project
LABOR_BILL_RATE_ORG_ID	NULL	NUMBER(15)	The identifier of the organization that owns the labor standard bill rate schedule
LABOR_SCHEDULE_FIXED_DATE	NULL	DATE	The fixed date used to determine the effective bill rates of the standard labor bill rate schedule. The project fixed date is a default for the task fixed date
LABOR_SCHEDULE_DISCOUNT	NULL	NUMBER(7,4)	The percentage to be discounted from the standard labor bill rate schedule. The project discount is a default for the task discount
NON_LABOR_STD_BILL_RATE_SCHDL	NULL	VARCHAR2(30)	The non-labor standard bill rate schedule that defaults to new tasks created for the project
NON_LABOR_BILL_RATE_ORG_ID	NULL	NUMBER(15)	The identifier of the organization that owns the non-labor standard bill rate schedule
NON_LABOR_SCHEDULE_FIXED_DATE	NULL	DATE	The fixed date used to determine the effective bill rates of the standard non-labor bill rate schedule. The project fixed date is a default for the task fixed date
NON_LABOR_SCHEDULE_DISCOUNT	NULL	NUMBER(7,4)	The percentage to be discounted from the standard non-labor bill rate schedule. The project discount is a default for the task discount
LIMIT_TO_TXN_CONTROLS_FLAG	NULL	VARCHAR2(1)	Flag that indicates if users can only charge expenditures to the project that are listed in the transaction controls. If flag =Y, then only items listed in the controls are allowed. If flag =N or null, then items not listed are allowed
PROJECT_LEVEL_FUNDING_FLAG	NULL	VARCHAR2(1)	Flag that indicates if the project is funded at the project level ('Y') or at the top task level ('N'). This flag is set when the project is baselined
INVOICE_COMMENT	NULL	VARCHAR2(240)	Free text comment that can be printed on an invoice. The invoice comment defined for the project is defaulted to the invoice comment for the draft invoices of the project

UNBILLED_RECEIVABLE_DR	NULL	NUMBER(22,5)	The unbilled receivables balance for the project that is posted to Oracle General Ledger or Oracle Receivables. This value is maintained by the Transfer Revenue and Transfer Invoices processes
UNEARNED_REVENUE_CR	NULL	NUMBER(22,5)	The unearned revenue balance for the project that is posted to Oracle General Ledger or Oracle Receivables. This value is maintained by the Transfer Revenue and Transfer Invoices processes
REQUEST_ID	NULL	NUMBER(15)	Standard Who Column
PROGRAM_ID	NULL	NUMBER(15)	Standard Who Column
PROGRAM_APPLICATION_ID	NULL	NUMBER(15)	Standard Who Column
PROGRAM_UPDATE_DATE	NULL	DATE	Standard Who Column
SUMMARY_FLAG	NOT NULL	VARCHAR2(1)	Flag used with the project key flexfield. The project key flexfield is not implemented
ENABLED_FLAG	NOT NULL	VARCHAR2(1)	Flag used with the key flexfield. The project key flexfield is not implemented
SEGMENT2	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT3	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT4	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT5	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT6	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT7	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT8	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT9	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
SEGMENT10	NULL	VARCHAR2(25)	Key flexfield segment - not implemented
ATTRIBUTE_CATEGORY	NULL	VARCHAR2(30)	Descriptive flexfield context field
ATTRIBUTE1	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE2	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE3	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE4	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE5	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE6	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE7	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE8	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE9	NULL	VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE10	NULL	VARCHAR2(150)	Descriptive flexfield segment
COST_IND_RATE_SCH_ID	NULL	NUMBER(15)	The identifier of default costing burden schedule
REV_IND_RATE_SCH_ID	NULL	NUMBER(15)	The identifier of default revenue burden schedule
INV_IND_RATE_SCH_ID	NULL	NUMBER(15)	The identifier of default invoice burden schedule
COST_IND_SCH_FIXED_DATE	NULL	DATE	The schedule fixed date of firm costing burden schedule

REV_IND_SCH_FIXED_DATE	NULL	DATE	The schedule fixed date of firm revenue burden schedule
INV_IND_SCH_FIXED_DATE	NULL	DATE	The schedule fixed date of firm invoice burden schedule
LABOR_SCH_TYPE	NULL	VARCHAR2(1)	The schedule type of labor expenditure items
NON_LABOR_SCH_TYPE	NULL	VARCHAR2(1)	The schedule type of non-labor expenditure items
OVR_COST_IND_RATE_SCH_ID	NULL	NUMBER(15)	This column is not used
OVR_REV_IND_RATE_SCH_ID	NULL	NUMBER(15)	This column is not used
OVR_INV_IND_RATE_SCH_ID	NULL	NUMBER(15)	This column is not used
TEMPLATE_FLAG	NULL	VARCHAR2(1)	Indicates whether the project is a Template or Not
VERIFICATION_DATE	NULL	DATE	Not currently used
CREATED_FROM_PROJECT_ID	NULL	NUMBER(15)	Holds the Project_id of the source project. This value would be null for templates
TEMPLATE_START_DATE_ACTIVE	NULL	DATE	The Effective start date of the Template
TEMPLATE_END_DATE_ACTIVE	NULL	DATE	Last effective date of the template
ORG_ID	NULL	NUMBER(15)	Operating unit identifier for multi-organization installations
PM_PRODUCT_CODE	NULL	VARCHAR2(30)	The identifier of the external project management system from which the project was imported.
PM_PROJECT_REFERENCE	NULL	VARCHAR2(25)	The identifier of the project in the external project management system from which the project was imported.
ACTUAL_START_DATE	NULL	DATE	The actual start date of the project. Applicable only for a project that has originated from an external system.
ACTUAL_FINISH_DATE	NULL	DATE	The actual end date of the project. Applicable only for a project that has originated from an external system.
EARLY_START_DATE	NULL	DATE	The early start date of the project. Applicable only for a project that has originated from an external system.
EARLY_FINISH_DATE	NULL	DATE	The early finish date of the project. Applicable only for a project that has originated from an external system.
LATE_START_DATE	NULL	DATE	The late start date of the project. Applicable only for a project that has originated from an external system.
LATE_FINISH_DATE	NULL	DATE	The late finish date of the project. Applicable only for a project that has originated from an external system.
SCHEDULED_START_DATE	NULL	DATE	The scheduled start date of the project. Applicable only for a project that has originated from an external system.
SCHEDULED_FINISH_DATE	NULL	DATE	The scheduled finish date of the project. Applicable only for a project that has originated from an external system.

BILLING_CYCLE_ID	NULL	NUMBER(15)	The identifier of the billing cycle which applies to the project
ADW_NOTIFY_FLAG	NULL	VARCHAR2(1)	The flag that indicates whether this row needs to be sent to the interface table or not. This column is required for Oracle Project Analysis Collection Pack.
WF_STATUS_CODE	NULL	VARCHAR2(30)	Indicates the status of the workflow associated with the project
OUTPUT_TAX_CODE	NULL	VARCHAR2(50)	Indicates whether tax rate defined for Projects will be used for Customer Invoices
RETENTION_TAX_CODE	NULL	VARCHAR2(50)	Indicates whether tax rate defined for retention will be used for Customer Invoices
PROJECT_CURRENCY_CODE	NOT NULL	VARCHAR2(15)	Currency code of the project
ALLOW_CROSS_CHARGE_FLAG	NOT NULL	VARCHAR2(1)	Flag to indicate whether cross charge are allowed
PROJECT_RATE_DATE	NULL	DATE	Default value for project rate date which can be overridden at task
PROJECT_RATE_TYPE	NULL	VARCHAR2(30)	Default value for project rate type which can be overridden at task
CC_PROCESS_LABOR_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates cross charge processing is to be performed for labor transactions charged to the project. Default value for the project template is N. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date.
LABOR_TP_SCHEDULE_ID	NULL	NUMBER	Identifier for transfer price schedule to use for cross charged labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date. If cc_process_labor_flag is set to Y, this field is required.
LABOR_TP_FIXED_DATE	NULL	DATE	Fixed date to find the effective rate of the bill rate or burden schedule when determining the transfer price for labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date.
CC_PROCESS_NL_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates cross charge processing is to be performed for n on-labor transactions charged to the project. Default value for the project template is N. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date.

NL_TP_SCHEDULE_ID	NULL NUMBER	Identifier for transfer price schedule to use for cross charged non-labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date. If cc_process_nl_flag is set to Y, this field is required.
NL_TP_FIXED_DATE	NULL DATE	Fixed date to find the effective rate of the bill rate or burden schedule when determining the transfer price for non-labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date.
CC_TAX_TASK_ID	NULL NUMBER	Identifier of the task to which intercompany tax items on the intercompany AP invoice are charged.
BILL_JOB_GROUP_ID	NULL NUMBER(15)	
COST_JOB_GROUP_ID	NULL NUMBER(15)	
ROLE_LIST_ID	NULL NUMBER(15)	Role_List is a list of allowable roles that are displayed when the Key/Team members are assigned.
WORK_TYPE_ID	NULL NUMBER(15)	Work_Types are predefined types of work, for example, Vacation, Training, and Administration.
CALENDAR_ID	NULL NUMBER(15)	Define a calendar with calendar exceptions primarily public holidays.
LOCATION_ID	NULL NUMBER(15)	Define a Project work site location.
PROBABILITY_MEMBER_ID	NULL NUMBER(15)	Project probability represents the probability of a project becoming approved. It is used as a weighting average for reporting.
PROJECT_VALUE	NULL NUMBER	Project value is used in reporting. For contract projects, it is the revenue amount. For capital and indirect projects, it is the cost amount. It is not assumed that the cost amount is raw or burdened.
EXPECTED_APPROVAL_DATE	NULL DATE	The expected date of the project approval. This is used for information purposes only.
RECORD_VERSION_NUMBER	NULL NUMBER(15)	Storing Locking sequence number.
INITIAL_TEAM_TEMPLATE_ID	NULL NUMBER(15)	
JOB_BILL_RATE_SCHEDULE_ID	NULL NUMBER	Stores the Job based Bill Rate Schedule ID for the project
EMP_BILL_RATE_SCHEDULE_ID	NULL NUMBER	Stores the Employee based Bill Rate Schedule ID for the project

### Indexes

Index Name	Index Type	Sequence	Column Name
PA_PROJECTS_N1	NOT UNIQUE	1	LABOR_BILL_RATE_ORG_ID
		2	LABOR_STD_BILL_RATE_SCHDL
PA_PROJECTS_N2	NOT UNIQUE	1	CARRYING_OUT_ORGANIZATION_ID
PA_PROJECTS_N3	NOT UNIQUE	1	NON_LABOR_BILL_RATE_ORG_ID
		2	NON_LABOR_STD_BILL_RATE_SCHDL
PA_PROJECTS_N4	NOT UNIQUE	1	PROJECT_STATUS_CODE
PA_PROJECTS_N5	NOT UNIQUE	2	ADW_NOTIFY_FLAG
PA_PROJECTS_N6	NOT UNIQUE	5	TEMPLATE_FLAG
PA_PROJECTS_U1	UNIQUE	1	PROJECT_ID
PA_PROJECTS_U2	UNIQUE	1	SEGMENT1
PA_PROJECTS_U3	UNIQUE	1	NAME
PA_PROJECTS_U4	UNIQUE	1	PM_PROJECT_REFERENCE
		2	PM_PRODUCT_CODE

### Sequences

Sequence	Derived Column
PA_PROJECTS_S	PROJECT_ID

### Database Triggers

Trigger Name : PA\_ADW\_PROJECTS\_T1  
Trigger Time : BEFORE  
Trigger Level : ROW  
Trigger Event : UPDATE

This trigger maintains the ADW\_NOTIFY\_COLUMN for any updates to PROJECT\_TYPE, NAME, SEGMENT1, DESCRIPTION and CARRYING\_OUT\_ORGANIZATION\_ID columns

## PA\_TASKS

PA\_TASKS stores user-defined subdivisions of project work. Tasks can be broken down into multiple levels of subtasks. The depth and width of the work breakdown structure (WBS) are unlimited. The WBS is held using the TASK\_ID and PARENT\_TASK\_ID columns. Two columns related to the WBS are denormalized: TOP\_TASK\_ID and WBS\_LEVEL.

### Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
HR_ALL_ORGANIZATION_UNITS	ORGANIZATION_ID	CARRYING_OUT_ORGANIZATION_ID
HR_ALL_ORGANIZATION_UNITS	ORGANIZATION_ID	CARRYING_OUT_ORGANIZATION_ID
PA_CC_TP_SCHEDULES_BG	TP_SCHEDULE_ID	LABOR_TP_SCHEDULE_ID
PA_CC_TP_SCHEDULES_BG	TP_SCHEDULE_ID	NL_TP_SCHEDULE_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	COST_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	INV_IND_RATE_SCH_ID
PA_IND_RATE_SCHEDULES_ALL_BG	IND_RATE_SCH_ID	REV_IND_RATE_SCH_ID
PA_LABOR_COST_MULTIPLIERS	LABOR_COST_MULTIPLIER_NAME	LABOR_COST_MULTIPLIER_NAME
PA_PROJECTS_ALL	PROJECT_ID	PROJECT_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	ORGANIZATION_ID	LABOR_BILL_RATE_ORG_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	STD_BILL_RATE_SCHEDULE	LABOR_STD_BILL_RATE_SCHDL
PA_STD_BILL_RATE_SCHEDULES_ALL	ORGANIZATION_ID	NON_LABOR_BILL_RATE_ORG_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	STD_BILL_RATE_SCHEDULE	NON_LABOR_STD_BILL_RATE_SCHDL
PA_STD_BILL_RATE_SCHEDULES_ALL	BILL_RATE_SCH_ID	JOB_BILL_RATE_SCHEDULE_ID
PA_STD_BILL_RATE_SCHEDULES_ALL	BILL_RATE_SCH_ID	EMP_BILL_RATE_SCHEDULE_ID
PA_TASKS	TASK_ID	PARENT_TASK_ID
PA_TASKS	TASK_ID	TOP_TASK_ID
PER_ALL_PEOPLE_F	PERSON_ID	TASK_MANAGER_PERSON_ID
RA_ADDRESSES_ALL_	ADDRESS_ID	ADDRESS_ID

### QuickCodes Columns

Column	QuickCodes Type	QuickCodes Table
LABOR_SCH_TYPE	PROJECT SCHEDULE TYPE	PA_LOOKUPS
	B	Bill Rate
	I	Indirect
NON_LABOR_SCH_TYPE	PROJECT SCHEDULE TYPE	PA_LOOKUPS
	B	Bill Rate
	I	Indirect
PM_PRODUCT_CODE	PM_PRODUCT_CODE	PA_LOOKUPS
SERVICE_TYPE_CODE	SERVICE TYPE	PA_LOOKUPS

### Column Descriptions

Name	Null?	Type	Description
TASK_ID (PK)	NOT NULL	NUMBER(15)	The system-generated number that uniquely identifies the task
PROJECT_ID	NOT NULL	NUMBER(15)	The identifier of the project for which the task is a unit of work

TASK_NUMBER	NOT NULL	VARCHAR2(25)	User-defined number that uniquely identifies the task within a project. It is recommended to number tasks based on the wbs since that is how tasks are ordered in some reports. However note: task numbers do not record the wbs structure
CREATION_DATE	NOT NULL	DATE	Standard Who Column
CREATED_BY	NOT NULL	NUMBER(15)	Standard Who Column
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who Column
LAST_UPDATED_BY	NOT NULL	NUMBER(15)	Standard Who Column
LAST_UPDATE_LOGIN	NOT NULL	NUMBER(15)	Standard Who Column
TASK_NAME	NOT NULL	VARCHAR2(20)	User-defined short name of the task
TOP_TASK_ID	NOT NULL	NUMBER(15)	The identifier of the top task to which this tasks rolls up. If the task is a top task, the top_task_id is set to its own task_id.
WBS_LEVEL	NOT NULL	NUMBER(3)	The level of the task in the work breakdown structure
READY_TO_BILL_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates whether the task is authorized to be invoiced. The flag is set only for top tasks, defaults to 'Y' upon creation of the task and is set to 'Y' or 'N' in the Control Revenue and Billing by Top Task form
READY_TO_DISTRIBUTE_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates whether the task is authorized for revenue accrual. The flag is set only for top tasks, defaults to 'Y' upon creation of the task, and is set to 'Y' or 'N' in the Control Revenue and Billing by Top Task form
PARENT_TASK_ID	NULL	NUMBER(15)	The identifier of the task that is the parent of the task in the project work breakdown structure
DESCRIPTION	NULL	VARCHAR2(250)	Description of the task
CARRYING_OUT_ORGANIZATION_ID	NOT NULL	NUMBER(15)	The identifier of the organization that is responsible for the task work. The project organization is defaulted to the task organization upon creation of the task
SERVICE_TYPE_CODE	NULL	VARCHAR2(30)	The type of work performed on the task
TASK_MANAGER_PERSON_ID	NULL	NUMBER(9)	The identifier of the employee that manages the task.
CHARGEABLE_FLAG	NULL	VARCHAR2(1)	Flag that indicates if expenditure items can be charged to the task. Only lowest level tasks can be chargeable

BILLABLE_FLAG	NULL VARCHAR2(1)	Default flag for items charged to the task that indicates if the item can accrue revenue (Y or N). For capital projects this flag is used as capitalizable_flag. For indirect projects this flag is set to N and is not used.
LIMIT_TO_TXN_CONTROLS_FLAG	NULL VARCHAR2(1)	Flag that indicates if users can only charge expenditures to the task that are listed in task's transaction controls. If flag =Y, then only items listed in controls are allowed. If flag = 'N' or null, then items not listed are allowed
START_DATE	NULL DATE	The date on which the task starts; expenditure items with item dates before the start date cannot be entered for the task
COMPLETION_DATE	NULL DATE	The date on which the task is completed; expenditure items with item dates after the task completion date cannot be entered for the task. You must enter a start date to enter a completion date.
ADDRESS_ID	NULL NUMBER(15)	The identifier of the customer address that is the task work site. This value is defaulted from the project customer work site if only one project customer exists
LABOR_BILL_RATE_ORG_ID	NULL NUMBER(15)	The identifier of the organization that owns the labor standard bill rate schedule
LABOR_STD_BILL_RATE_SCHDL	NULL VARCHAR2(20)	The labor standard bill rate schedule that is used to calculate revenue for labor expenditure items charged to the task
LABOR_SCHEDULE_FIXED_DATE	NULL DATE	The date used to determine the effective bill rates of the task standard labor bill rate schedule. If no fixed date is entered, the expenditure item date is used to determine the effective bill rate for the item
LABOR_SCHEDULE_DISCOUNT	NULL NUMBER(7,4)	The percentage to be discounted from the task standard labor bill rate schedule
NON_LABOR_BILL_RATE_ORG_ID	NULL NUMBER(15)	The identifier of the organization that owns the non-labor standard bill rate schedule
NON_LABOR_STD_BILL_RATE_SCHDL	NULL VARCHAR2(30)	The non-labor standard bill rate schedule that is used to calculate revenue for non-labor expenditure items charged to the task

NON_LABOR_SCHEDULE_FIXED_DATE	NULL DATE	The fixed date used to determine the effective bill rates of the standard non-labor bill rate schedule. If no fixed date is entered, the expenditure item date is used to determine the effective bill rate of the item
NON_LABOR_SCHEDULE_DISCOUNT	NULL NUMBER(7,4)	The percentage to be discounted from the task standard non-labor bill rate schedule
LABOR_COST_MULTIPLIER_NAME	NULL VARCHAR2(20)	The labor cost multiplier defined for the task of a premium project. The labor cost multiplier is populated for all overtime expenditure items charged to the task upon manual entry of the items or in the Overtime Calculation program
REQUEST_ID	NULL NUMBER(15)	Standard Who Column
PROGRAM_APPLICATION_ID	NULL NUMBER(15)	Standard Who Column
PROGRAM_ID	NULL NUMBER(15)	Standard Who Column
PROGRAM_UPDATE_DATE	NULL DATE	Standard Who Column
ATTRIBUTE_CATEGORY	NULL VARCHAR2(30)	Descriptive flexfield context field
ATTRIBUTE1	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE2	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE3	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE4	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE5	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE6	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE7	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE8	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE9	NULL VARCHAR2(150)	Descriptive flexfield segment
ATTRIBUTE10	NULL VARCHAR2(150)	Descriptive flexfield segment
COST_IND_RATE_SCH_ID	NULL NUMBER(15)	The identifier of default costing burden schedule
REV_IND_RATE_SCH_ID	NULL NUMBER(15)	The identifier of default revenue burden schedule
INV_IND_RATE_SCH_ID	NULL NUMBER(15)	The identifier of default invoice burden schedule
COST_IND_SCH_FIXED_DATE	NULL DATE	The schedule fixed date of firm costing burden schedule
REV_IND_SCH_FIXED_DATE	NULL DATE	The schedule fixed date of firm revenue burden schedule
INV_IND_SCH_FIXED_DATE	NULL DATE	The schedule fixed date of firm invoice burden schedule
LABOR_SCH_TYPE	NULL VARCHAR2(1)	The schedule type of labor expenditure items
NON_LABOR_SCH_TYPE	NULL VARCHAR2(1)	The schedule type of non-labor expenditure items
OVR_COST_IND_RATE_SCH_ID	NULL NUMBER(15)	This column is not used
OVR_INV_IND_RATE_SCH_ID	NULL NUMBER(15)	This column is not used
OVR_REV_IND_RATE_SCH_ID	NULL NUMBER(15)	This column is not used
PM_PRODUCT_CODE	NULL VARCHAR2(30)	The identifier of the external project management system from which the task was imported.
PM_TASK_REFERENCE	NULL VARCHAR2(25)	The identifier of the task in the external project management system from which the budget was imported.

ACTUAL_START_DATE	NULL	DATE	The actual start date of the project. Applicable only for a project that has originated from an external system.
ACTUAL_FINISH_DATE	NULL	DATE	The actual end date of the project. Applicable only for a project that has originated from an external system.
EARLY_START_DATE	NULL	DATE	The early start date of the project. Applicable only for a project that has originated from an external system.
EARLY_FINISH_DATE	NULL	DATE	The early finish date of the project. Applicable only for a project that has originated from an external system.
LATE_START_DATE	NULL	DATE	The late start date of the project. Applicable only for a project that has originated from an external system.
LATE_FINISH_DATE	NULL	DATE	The late finish date of the project. Applicable only for a project that has originated from an external system.
SCHEDULED_START_DATE	NULL	DATE	The scheduled start date of the project. Applicable only for a project that has originated from an external system.
SCHEDULED_FINISH_DATE	NULL	DATE	The scheduled finish date of the project. Applicable only for a project that has originated from an external system.
ADW_NOTIFY_FLAG	NULL	VARCHAR2(1)	The flag that indicates whether this row needs to be sent to the interface table or not. This column is required for Oracle Project Analysis Collection Pack.
ALLOW_CROSS_CHARGE_FLAG	NOT NULL	VARCHAR2(1)	Flag to indicate whether cross charge are allowed
PROJECT_RATE_DATE	NULL	DATE	Task level default value for project rate date
PROJECT_RATE_TYPE	NULL	VARCHAR2(30)	Task level default value for project rate type
CC_PROCESS_LABOR_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates cross charge processing is to be performed for labor transactions charged to the project. Default value for the project template is N. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date
LABOR_TP_SCHEDULE_ID	NULL	NUMBER	Identifier for transfer price schedule to use for cross charged labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date. If cc_process_labor_flag is set to Y, this field is required

LABOR_TP_FIXED_DATE	NULL	DATE	Fixed date to find the effective rate of the bill rate or burden schedule when determining the transfer price for labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date
CC_PROCESS_NL_FLAG	NOT NULL	VARCHAR2(1)	Flag that indicates cross charge processing is to be performed for non-labor transactions charged to the project. Default value for the project template is N. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date
NL_TP_SCHEDULE_ID	NULL	NUMBER	Identifier for transfer price schedule to use for cross charged non-labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date. If cc_process_nl_flag is set to Y, this field is required
NL_TP_FIXED_DATE	NULL	DATE	Fixed date to find the effective rate of the bill rate or burden schedule when determining the transfer price for non-labor transactions. This is defaulted to a project from the project template. This value for the project is a default for the task fixed date
RECEIVE_PROJECT_INVOICE_FLAG	NULL	VARCHAR2(1)	Flag that indicates that the task may receive charges from internal supplies via inter-project billing
WORK_TYPE_ID	NULL	NUMBER(15)	Work_Types are predefined types of work, for example,
RECORD_VERSION_NUMBER	NULL	NUMBER(15)	Stores locking sequence number
JOB_BILL_RATE_SCHEDULE_ID	NULL	NUMBER	Stores the job based bill rate schedule ID for the project.
EMP_BILL_RATE_SCHEDULE_ID	NULL	NUMBER	Stores the employee based bill rate schedule ID for the project.

*Indexes*

Index Name	Index Type	Sequence	Column Name
PA_TASKS_N1	NOT UNIQUE	1	LABOR_BILL_RATE_ORG_ID
		2	LABOR_STD_BILL_RATE_SCHDL
PA_TASKS_N10	NOT UNIQUE	5	ADDRESS_ID
PA_TASKS_N2	NOT UNIQUE	1	CARRYING_OUT_ORGANIZATION_ID
PA_TASKS_N3	NOT UNIQUE	1	SERVICE_TYPE_CODE
PA_TASKS_N4	NOT UNIQUE	1	PARENT_TASK_ID
PA_TASKS_N5	NOT UNIQUE	1	NON_LABOR_BILL_RATE_ORG_ID
		2	NON_LABOR_STD_BILL_RATE_SCHDL
PA_TASKS_N6	NOT UNIQUE	1	TASK_MANAGER_PERSON_ID
PA_TASKS_N7	NOT UNIQUE	1	TOP_TASK_ID
PA_TASKS_N8	NOT UNIQUE	2	PROJECT_ID
		4	PM_TASK_REFERENCE

**Oracle Proprietary, Confidential Information--Use Restricted by Contract**

PA_TASKS_N9	NOT UNIQUE	2	ADW_NOTIFY_FLAG
PA_TASKS_U1	UNIQUE	1	TASK_ID
PA_TASKS_U2	UNIQUE	1	PROJECT_ID
		2	TASK_NUMBER

*Sequences*

<u>Sequence</u>	<u>Derived Column</u>
PA_TASKS_S	TASK_ID

*Database Triggers*

Trigger Name : PA\_ADW\_TASKS\_T1  
 Trigger Time : BEFORE  
 Trigger Level : ROW  
 Trigger Event : UPDATE

This trigger maintains the ADW\_NOTIFY\_FLAG column for any updates to TOP\_TASK\_ID, DESCRIPTION, TASK\_NUMBER, TASK\_NAME, CARRYING\_OUT\_ORGANIZATION\_ID, SERVICE\_TYPE\_CODE and PROJECT\_ID columns. If the service\_type\_code or the carrying\_out\_organization\_id is changed then it creates a record in PA\_TASK\_HISTORY table also.

Trigger Name : PA\_ADW\_TASKS\_T2  
 Trigger Time : BEFORE  
 Trigger Level : ROW  
 Trigger Event : INSERT

This trigger creates a new record into the PA\_TASK\_HISTORY table for new records created in PA\_TASKS table.





## **A**

Application Building Block, 1 – 18

## **D**

Database Diagram, 1 – 18

Summary Database Diagram, 1 – 18

## **F**

Form, 1 – 19

## **M**

Modules, 1 – 18

## **Q**

QuickCodes, 1 – 19

## **R**

Relationship, 1 – 18





## Reader's Comment Form

### Oracle Product Development Exchange Technical Reference Manual, A88861-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information we use for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual? What did you like least about it?

If you find any errors or have any other suggestions for improvement, please indicate the topic, chapter, and page number below:

---

---

---

---

---

---

---

---

---

---

Please send your comments to:

Oracle Applications Documentation Manager  
Oracle Corporation  
500 Oracle Parkway  
Redwood Shores, CA 94065 USA  
Phone: (650) 506-7000 Fax: (650) 506-7200

If you would like a reply, please give your name, address, and telephone number below:

---

---

---

Thank you for helping us improve our documentation.

L