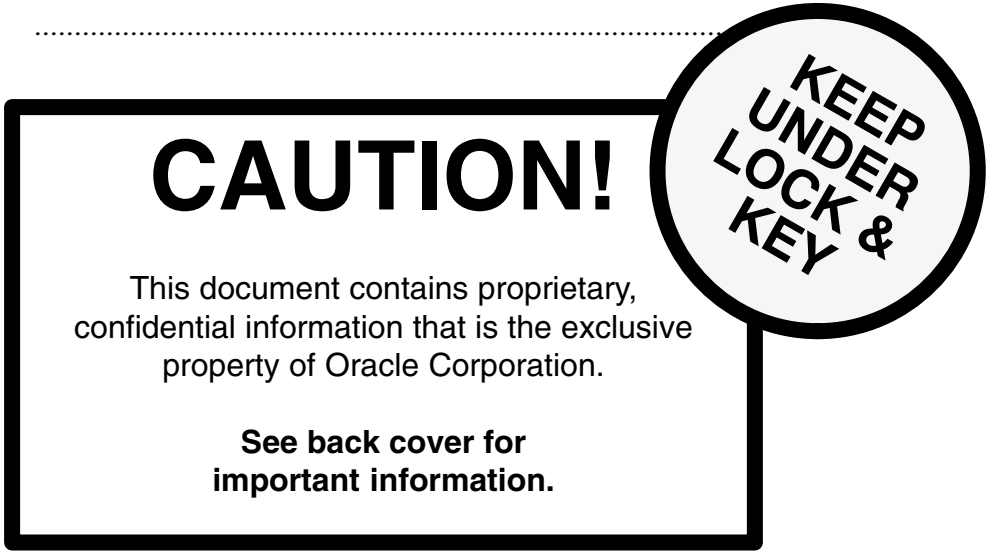
Oracle® Applications Implementation

Wizard

Technical Reference Manual

Release 11*i*



Oracle® Applications Implementation Wizard Technical Reference Manual

RELEASE 11*i*

January 2000

Oracle® Applications Implementation Wizard Technical Reference Manual
Release 11*i*

To order this book, ask for Part No. A80906–01

Major Contributors: Gail D'Aloisio, Manoj Gupta, Jie Ke

Contributors: Christopher Andrews

CAUTION

This 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.

Contents

CHAPTER

1 Introduction

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

This chapter introduces you to the *Oracle Applications Implementation Wizard 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 Applications Implementation Wizard Technical Reference Manual* contains detailed, up–to–date information about the underlying structure of the Implementation Wizard. As we design and build new releases of the Implementation Wizard, we update our Oracle Designer repository to reflect our enhancements. As a result, we can always provide you with an *Oracle Applications Implementation Wizard 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 the Implementation Wizard to improve performance.

About this Manual

This manual describes the Oracle Applications Release 11i data model, as used by the Implementation Wizard; 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 the Implementation Wizard 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 the Implementation Wizard. You can also use Oracle*MetaLink* which is accessible through Oracle's Support Web Center (*http://www.oracle.com/support/elec_sup*).

Finding the Latest Information

The *Oracle Applications Implementation Wizard Technical Reference Manual* contains the latest information as of the publication date. For the latest information we encourage you to use Oracle*MetaLink* which is accessible through Oracle's Support Web Center (*http://www.oracle.com/support/elec_sup*).

Audience

The *Oracle Applications Implementation Wizard 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 the Implementation Wizard. 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 the Implementation Wizard uses. This chapter also has a list of modules.

Detailed Design

This section, Chapter 3, contains a detailed description of the Implementation Wizard 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 Applications Implementation Wizard Technical Reference Manual* is a single, centralized source for all the information you need to know about the underlying structure and processing of the Implementation Wizard. For example, you can use this manual when you need to:

- Convert existing application data
- Integrate your Implementation Wizard 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 the Implementation Wizard. Modifying the Implementation Wizard limits your ability to upgrade to future releases of your Implementation Wizard application. In addition, it interferes with our ability to give you the high–quality support you deserve.

We have constructed the Implementation Wizard 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 Applications Implementation Wizard Technical Reference Manual* does not contain complete information about the dependencies between Implementation Wizard 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 Implementation Wizard 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 the Implementation Wizard 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 Implementation Wizard 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 Applications Implementation Wizard 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.

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 the Implementation Wizard.

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 the Implementation Wizard. 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*

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.

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

Oracle Global Financial Applications Technical Reference Manual

This manual describes additional tables and views that have been added to support features that meet statutory requirements and common business practices in your country or region. The Appendix of this manual also describes how the GLOBAL_ATTRIBUTE columns of your base product tables are used in your country or region.

Country–Specific Manuals

Use these manuals to meet statutory requirements and common business practices in your country or region. They also describe additional features added to the Implementation Wizard to meet those requirements. Look for a user guide appropriate to your country; for example, see the *Oracle Financials for the Czech Republic User Guide* for more information about using this software in the Czech Republic.

Oracle Self–Service Web Applications Online Documentation

This documentation describes how Oracle Self–Service Web Applications enable companies to provide a self–service and secure Web interface for employees, customers, and suppliers. Employees can change their personal status, submit expense reports, or request supplies. Customers can check on their orders, and suppliers can share production schedules with their trading partners. This documentation is available in HTML only.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Implementation Wizard 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 the Implementation Wizard. This manual details additional steps and setup considerations for implementing the Implementation Wizard with this feature.

Oracle Financials Open Interfaces Guide

This book contains a brief summary of all open interface discussions in all Oracle Financial Applications user guides.

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, 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 the Implementation Wizard 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 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 the Implementation Wizard 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 the Implementation Wizard 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.

Or, send electronic mail to **appsdoc@us.oracle.com**.

CHAPTER

2 High–Level Design

This chapter presents a high–level design for the Implementation Wizard that satisfies the business needs we specify during Strategy and Analysis. It contains database diagrams for the Implementation Wizard 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 Implementation Wizard applications. And, you can prepare yourself to understand the detailed design and implementation of the Implementation Wizard.

Summary Database Diagram

The Summary Database Diagram section graphically represents 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.

We prepare a summary database diagram to describe, at a conceptual level, the key information on which our business depends. Later, we refine this summary database diagram, breaking it into multiple database diagrams (generally, one per application building block) to represent all the tables and relationships we need to implement our application in the database.

Review the Summary Database Diagram section to see at a glance the major tables and relationships on which the Implementation Wizard application depends.

Database Diagrams

The Database Diagrams section graphically represents all Implementation Wizard applications tables and the relationships between them, organized by building block.

Use this section to quickly learn what tables each Implementation Wizard 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 Implementation Wizard 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 the Implementation Wizard; we do not provide additional documentation for these tables.

View Lists

The View List sections list the Implementation Wizard 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 the Implementation Wizard 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 Implementation Wizard 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 the Implementation Wizard. Use your user's guide to learn more about reports and concurrent processes.

Summary Database Diagram

This Summary Database Diagram graphically represents the most important Implementation Wizard applications tables and the relationships between them. It describes, at a conceptual level, the key information on which your Implementation Wizard application depends.

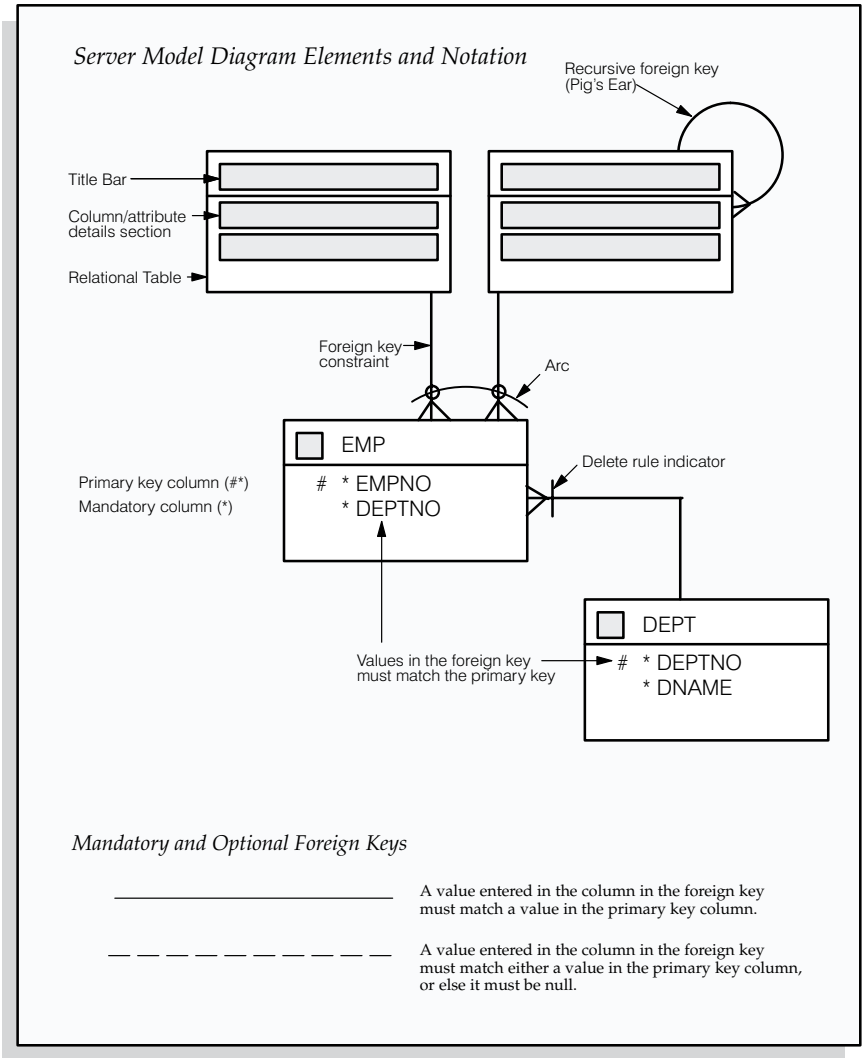
This diagram does not represent the complete database implementation of Implementation Wizard applications tables. It shows tables that contain key reference and transaction data, and omits tables and relationships that contribute little to the understanding of the Implementation Wizard applications data model. For example, a foreign key relationship shown between two tables may actually be implemented by an intervening table, not shown in this diagram.

For more detailed graphical representations of Implementation Wizard applications tables and the relationships between them, see the Database Diagrams section in this chapter.

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.

Public Table List

This section lists each public database table that the Applications Implementation Wizard 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.

The Application Implementation Wizard uses the following Public tables:

Table Name	Description
AZ_ARCHIVE	Stores wizard data (See page 3 – 8)
AZ_ARCHIVE_STEPS	Stores Wizard data pertaining to tasks (See page 3 – 9)
AZ_GROUPS	Process groups information (See page 3 – 11)
AZ_PROCESSES	Workflow processes information (See page 3 – 15)
AZ_PRODUCT_FLOWS	Workflow processes information for each product (See page 3 – 19)
AZ_PRODUCT_PHASES	Phase information for each installed product (See page 3 – 20)

Public View List

This section lists each public database view that the Implementation Wizard 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.

Implementation Wizard uses the following public views:

View Name	Description
AZ_FLOW_PHASES_V	Provides easy access to Wizard workflows and their phase (See page 3 – 10)
AZ_GROUPS_V	Provides easy access to language–dependent display names for Wizard process groups (See page 3 – 13)
AZ_PROCESSES_ALL_V	Provides easy access to language–dependent display names as well as statuses for all Wizard processes (See page 3 – 16)
AZ_PROCESSES_V	Selects all processes for the current implementation path (See page 3 – 17)
AZ_PROCESS_ACTIVITIES_V	Stores the definition of a Wizard process (See page 3 – 18)
AZ_PRODUCT_PHASES_V	Stores language–dependent application names along with their phase information (See page 3 – 21)
AZ_TASKS_V	Provides easy access to assigned use and status, as well as context names for Wizard tasks. (See page 3 – 22)

Module List

This section lists each form, report and concurrent program comprising Application Implementation/Wizard.

Forms

AZWIZARD	Implementation Wizard
AZWMODES	Define Implementation Mode
AZWPHASE	Define Phase

CHAPTER

3 Detailed Design

This chapter presents a detailed design for implementing Implementation Wizard. 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 Implementation Wizard that enables you to:

- Convert existing application data
- Integrate your Implementation Wizard 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 Implementation Wizard 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 Implementation Wizard 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 Implementation Wizard 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

Because Oracle does not support customization of Oracle Application Object Library tables, we do not provide you with detailed information about them. Consequently, this section does not document all the FND_% tables Implementation Wizard uses.

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 Implementation Wizard 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 Implementation Wizard 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:

Not currently used	Implementation Wizard does not use this column, although the column might be used in a future release.
No longer used	Implementation Wizard no longer uses this column. AutoInstall installs this column. Subsequent versions of Implementation Wizard might not include this column.
No longer installed	Implementation Wizard no longer uses this column. If you *upgraded* 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 *install* Implementation Wizard, you do not have this column.

Standard Who Columns

Most Implementation Wizard tables contain standard columns to support \ **R**ow **W**ho. 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
CREATED_BY	User who created this row (foreign key to FND_USER.USER_ID)
LAST_UPDATE_LOGIN	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 Implementation Wizard 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:

REQUEST_ID	Concurrent request ID of program that last updated this row (foreign key to FND_CONCURRENT_REQUESTS.REQUEST_ID)
PROGRAM_APPLICATION_ID	Application ID of program that last updated this row (foreign key to FND_APPLICATION.APPLICATION_ID)
PROGRAM_ID	Program ID of program that last updated this row (foreign key to FND_CONCURRENT_PROGRAM.CONCURRENT_PROGRAM_ID)
PROGRAM_UPDATE_DATE	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 Localizations

Some tables have GLOBAL_ATTRIBUTE columns which support additional features added to Implementation Wizard 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 Regional 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 Implementation Wizard 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 Implementation Wizard.

Sequences

Implementation Wizard 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 it fires.

View Derivation

For each Implementation Wizard 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.

AZ_ARCHIVE

AZ_ARCHIVE stores Wizard data pertaining to groups, processes, and tasks. Each row in the table includes the node ID, the node type that indicates whether it is a group, process, or task, the node name that is the label for the process or group, the context type for the process, the context name for the process, the context ID for the process, the status code indicating if it is active or complete, the item key to identify the task, the assigned user for the task role, and comments for the process.

Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
AZ_ARCHIVE	NODE_ID	PARENT_NODE_ID

Column Descriptions

Name	Null?	Type	Description
NODE_ID (PK)	NOT NULL	VARCHAR2(240)	Unique identifier
NODE_TYPE	NOT NULL	VARCHAR2(1)	Indicates if it is group, process, or task
NODE_NAME	NULL	VARCHAR2(80)	Label for group, process, or task
CONTEXT_TYPE	NULL	VARCHAR2(30)	Context type of the process (BG for business group, IO for inventory organization, OU for operating unit, SOB for set of books)
CONTEXT_TYPE_NAME	NULL	VARCHAR2(80)	Context name for process (Business Group for BG, Set of Books for SOB, Operating Unit for OU, Inventory Organization for IO)
CONTEXT_ID	NULL	NUMBER(15)	Context identifier for the process
CONTEXT_NAME	NULL	VARCHAR2(80)	Context name
STATUS_CODE	NULL	VARCHAR2(30)	Status for a process or task (A for active, C for complete, and N for not started process)
STATUS_NAME	NULL	VARCHAR2(80)	Label or meaning for status codes
ITEM_KEY	NULL	VARCHAR2(240)	Unique identifier for task
ASSIGNED_USER	NULL	VARCHAR2(100)	Role for the task
ASSIGNED_USER_NAME	NULL	VARCHAR2(4000)	User name for the task
START_DATE	NULL	DATE	Start date for the task
END_DATE	NULL	DATE	Date when the task was completed
DURATION	NULL	NUMBER(15)	Time for which a task was active
COMMENTS	NULL	VARCHAR2(2000)	User-entered comments for process
PARENT_NODE_ID	NULL	VARCHAR2(240)	Parent identifier

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_ARCHIVE_N1	NOT UNIQUE	1	PARENT_NODE_ID
AZ_ARCHIVE_U1	UNIQUE	1	NODE_ID

AZ_ARCHIVE_STEPS

AZ_ARCHIVE_STEPS stores Wizard data regarding tasks and all their steps. Each row has information regarding each step of the task, its status, the assigned user, the form name associated with the step, and the begin date for the the task step.

Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
AZ_ARCHIVE	NODE_ID	NODE_ID

Column Descriptions

Name	Null?	Type	Description
ITEM_KEY (PK)	NOT NULL	VARCHAR2(240)	Unique identifier for the task
ACTIVITY_TYPE (PK)	NOT NULL	VARCHAR2(8)	Identifies type of task step (FUNCTION, NOTICE, or PROCESS)
STEP (PK)	NOT NULL	VARCHAR2(30)	Step for the task
STEP_NAME	NOT NULL	VARCHAR2(80)	Label for the step
ASSIGNED_USER	NULL	VARCHAR2(100)	Assigned role for notification
ASSIGNED_USER_NAME	NULL	VARCHAR2(4000)	User name for the notification
FORM_NAME	NULL	VARCHAR2(4000)	The form associated with the task step
STATUS_CODE	NULL	VARCHAR2(8)	Status for the task step, either ACTIVE, COMPLETE, NOTIFIED, or ERROR
RESULT_CODE	NULL	VARCHAR2(30)	Response code for the step (e.g. skipped,. done, etc.)
BEGIN_DATE (PK)	NOT NULL	DATE	Start date for the task step
END_DATE	NULL	DATE	End date for the task step
DURATION	NULL	NUMBER(15)	Duration of the task
COMMENTS	NULL	VARCHAR2(4000)	Comments
NODE_ID	NOT NULL	VARCHAR2(240)	Unique identifier for the task step

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_ARCHIVE_STEPS_N1	NOT UNIQUE	1	NODE_ID
AZ_ARCHIVE_STEPS_U1	UNIQUE	1	ITEM_KEY
		2	ACTIVITY_TYPE
		3	STEP
		4	BEGIN_DATE

AZ_FLOW_PHASES_V

AZ_FLOW_PHASES_V is a view that stores information about the workflow processes and their corresponding phase(s). Each row includes the item type, process name and its associated phase. This view is primarily used for backend processing.

View Definition

```
CREATE VIEW AZ_FLOW_PHASES_V
  as SELECT
     distinct azpf.item_type, azpf.process_name, azpp.phase


   where azpf.application_id = azpp.application_id
```

Column Descriptions

Name	Null?	Type	Description
ITEM_TYPE	NULL	VARCHAR2(8)	Refer to base table.
PROCESS_NAME	NULL	VARCHAR2(30)	Refer to base table.
PHASE	NULL	NUMBER(15)	Refer to base table.

AZ_GROUPS

AZ_GROUPS stores information about the process groups used in the Implementation Wizard form. Each record contains the unique identifier of the group, the parent process group that is associated with the group, as well as the display order, color code, complete flag, and the application or lookup code that the process group is associated with. Additionally, to enable different types of implementation of the Wizard, there is also a process type for each group indicating its implementation type.

Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
AZ_GROUPS	GROUP_ID PROCESS_TYPE	HIERARCHY_PARENT_ID HIERARCHY_PARENT_PROCESS_TYPE
AZ_GROUPS	GROUP_ID PROCESS_TYPE	DEPENDENCY_PARENT_ID DEPENDENCY_PARENT_PROCESS_TYPE

Column Descriptions

Name	Null?	Type	Description
GROUP_ID (PK)	NOT NULL	VARCHAR2(240)	Group identifier
PROCESS_TYPE (PK)	NOT NULL	VARCHAR2(30)	Identifies the implementation type for the process
HIERARCHY_PARENT_PROCESS_TYPE	NULL	VARCHAR2(30)	Implementation type for the parent of the node
COMPLETE_FLAG	NOT NULL	VARCHAR2(1)	Indicates whether the process group is complete
COLOR_CODE	NULL	VARCHAR2(30)	Color code associated with the process group
CREATION_DATE	NOT NULL	DATE	Standard Who column
CREATED_BY	NOT NULL	NUMBER(15)	Standard Who column
DEPENDENCY_PARENT_PROCESS_TYPE	NULL	VARCHAR2(30)	Implementation type (upgrade or fresh implementation) of dependency parent or node
DISPLAY_ORDER	NOT NULL	NUMBER(15)	Display order of process groups at different levels
APPLICATION_ID	NULL	NUMBER(15)	Application identifier
LOOKUP_CODE	NULL	VARCHAR2(30)	Quickcode code
LAST_UPDATE_DATE	NOT NULL	DATE	Standard Who column
LAST_UPDATED_BY	NOT NULL	NUMBER(15)	Standard Who column
LAST_UPDATE_LOGIN	NULL	NUMBER(15)	Standard Who column
HIERARCHY_PARENT_ID	NULL	VARCHAR2(240)	Parent process group that the process group is associated with in the hierarchy tree
DEPENDENCY_PARENT_ID	NULL	VARCHAR2(240)	Parent process group that the process group is associated with in the dependency chart

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_GROUPS_N1	NOT UNIQUE	1 2	HIERARCHY_PARENT_ID HIERARCHY_PARENT_PROCESS_TYPE
AZ_GROUPS_N2	NOT UNIQUE	1 2	DEPENDENCY_PARENT_ID DEPENDENCY_PARENT_PROCESS_TYPE

Indexes (Continued)

Index Name	Index Type	Sequence	Column Name
AZ_GROUPS_U1	UNIQUE	1	GROUP_ID
		2	PROCESS_TYPE

AZ_GROUPS_V

AZ_GROUPS_V is a view that joins AZ_GROUPS, FND_APPLICATION_VL, and FND_LOOKUPS to provide quick access to the display name of each group.

View Definition

```
    CREATE VIEW AZ_GROUPS_V
      as SELECT
         ag.group_id,
    ag.process_type,
    fl.meaning display_name,
    ag.display_order,
    decode(ag.complete_flag, 'N', 'A', 'C') status,
    ag.complete_flag,
    ag.application_id,
    ag.lookup_code,
    ag.color_code,
    ag.hierarchy_parent_id,
    ag.hierarchy_parent_process_type,
    ag.dependency_parent_id,
    ag.dependency_parent_process_type,
    ag.creation_date,
    ag.created_by,
    ag.last_update_date,
    ag.last_updated_by,
    ag.last_update_login

       from az_groups ag,
              fnd_lookups fl
    where ag.lookup_code IS NOT NULL
    and fl.lookup_type = 'AZ_PROCESS_GROUPS'
    and fl.lookup_code = ag.lookup_code
    union
    select
    ag.group_id,
    ag.process_type,
    fav.application_name display_name,
    ag.display_order,
    decode(ag.complete_flag, 'N', 'A', 'C') status,
    ag.complete_flag,
    ag.application_id,
    ag.lookup_code,
    ag.color_code,
    ag.hierarchy_parent_id,
    ag.hierarchy_parent_process_type,
    ag.dependency_parent_id,
    ag.dependency_parent_process_type,
    ag.creation_date,
    ag.created_by,
    ag.last_update_date,
    ag.last_updated_by,
    ag.last_update_login
```

```
from az_groups ag,
          fnd_application_vl fav
where ag.application_id IS NOT NULL
and ag.application_id = fav.application_id
```

Column Descriptions

Name	Null?	Type	Description
GROUP_ID	NULL	VARCHAR2(240)	Refer to base table.
PROCESS_TYPE	NULL	VARCHAR2(30)	Refer to base table.
DISPLAY_NAME	NULL	VARCHAR2(240)	Refer to base table.
DISPLAY_ORDER	NULL	NUMBER(15)	Refer to base table.
STATUS	NULL	VARCHAR2(1)	Refer to base table.
COMPLETE_FLAG	NULL	VARCHAR2(1)	Refer to base table.
APPLICATION_ID	NULL	NUMBER(15)	Refer to base table.
LOOKUP_CODE	NULL	VARCHAR2(30)	Refer to base table.
COLOR_CODE	NULL	VARCHAR2(30)	Refer to base table.
HIERARCHY_PARENT_ID	NULL	VARCHAR2(240)	Refer to base table.
HIERARCHY_PARENT_PROCESS_TYPE	NULL	VARCHAR2(30)	Refer to base table.
DEPENDENCY_PARENT_ID	NULL	VARCHAR2(240)	Refer to base table.
DEPENDENCY_PARENT_PROCESS_TYPE	NULL	VARCHAR2(30)	Refer to base table.
CREATION_DATE	NULL	DATE	Refer to base table.
CREATED_BY	NULL	NUMBER(15)	Refer to base table.
LAST_UPDATE_DATE	NULL	DATE	Refer to base table.
LAST_UPDATED_BY	NULL	NUMBER(15)	Refer to base table.
LAST_UPDATE_LOGIN	NULL	NUMBER(15)	Refer to base table.

AZ_PROCESSES

AZ_PROCESSES stores information about the runnable workflow processes with different context names and context types. Each record contains the item type, process name, context ID, context name, and context type, as well as the complete flag, the parent process group that it is associated with, the display order in the hierarchy tree, and the user entered comments for the process from the Implementation Wizard form.

Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
AZ_GROUPS	GROUP_ID	PARENT_ID
	PROCESS_TYPE	PROCESS_TYPE

Column Descriptions

Name	Null?	Type	Description
PROCESS_TYPE	NOT NULL	VARCHAR2(30)	Identifies if the process belongs to a fresh implementation or an upgrade
ITEM_TYPE (PK)	NOT NULL	VARCHAR2(30)	Item type
PROCESS_NAME (PK)	NOT NULL	VARCHAR2(30)	Internal name of the process activity
CONTEXT_ID (PK)	NOT NULL	NUMBER(15)	Context identifier
DISPLAY_ORDER	NOT NULL	NUMBER(15)	Display order of the process in the hierarchy tree
STATUS_CODE	NOT NULL	VARCHAR2(30)	Status for the process (A for Active, N for Not Started, C for Complete)
CONTEXT_TYPE	NOT NULL	VARCHAR2(30)	Context type of the process (BG for business group, IO for inventory organization, OU for operating unit, SOB for set of books)
CONTEXT_NAME	NULL	VARCHAR2(80)	Context name of the process
COMMENTS	NULL	VARCHAR2(2000)	User-entered comments for the process
PARENT_ID	NOT NULL	VARCHAR2(240)	Parent process group that the process is associated with
COMPLETE_FLAG	NULL	VARCHAR2(1)	Flag to indicate whether the process is complete

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_PROCESSES_N1	NOT UNIQUE	1	PARENT_ID
		2	PROCESS_TYPE
AZ_PROCESSES_U1	UNIQUE	1	ITEM_TYPE
		2	PROCESS_NAME
		3	CONTEXT_ID

AZ_PROCESSES_ALL_V

AZ_PROCESSES_V is a view based on ap_processes regardless of the process_type. It also provides quick access to the display name and the context type name of the processes.

View Definition

```
    CREATE VIEW AZ_PROCESSES_ALL_V
      as SELECT
         ap.item_type,ap.process_name,wav.display_name,
    ap.context_id,ap.parent_id,ap.display_order,ap.status_code,
    ap.process_type,
    ap.context_type,
     fl.meaning context_type_name,
    ap.context_name,
    ap.comments, wav.description

       from         az_processes ap,
                    wf_activities_vl wav,
                    fnd_lookups fl
    where   ap.item_type = wav.item_type
    and     ap.process_name = wav.name
    and           wav.item_type like 'AZ%'
    and           wav.name like 'AZ%'
    and     wav.end_date IS NULL
    and     fl.lookup_type = 'AZ_CONTEXT_TYPE'
    and     fl.lookup_code = ap.context_type
```

Column Descriptions

Name	Null?	Type	Description
ITEM_TYPE	NULL	VARCHAR2(8)	Refer to base table.
PROCESS_NAME	NULL	VARCHAR2(30)	Refer to base table.
DISPLAY_NAME	NULL	VARCHAR2(80)	Refer to base table.
CONTEXT_ID	NULL	NUMBER(15)	Refer to base table.
PARENT_ID	NULL	VARCHAR2(240)	Refer to base table.
DISPLAY_ORDER	NULL	NUMBER(15)	Refer to base table.
STATUS	NULL	VARCHAR2(15)	Refer to base table.
PROCESS_TYPE	NULL	VARCHAR2(15)	Refer to base table.
CONTEXT_TYPE	NULL	VARCHAR2(15)	Refer to base table.
CONTEXT_TYPE_NAME	NULL	VARCHAR2(80)	Refer to base table.
CONTEXT_NAME	NULL	VARCHAR2(80)	Refer to base table.
COMMENTS	NULL	VARCHAR2(2000)	Refer to base table.
DESCRIPTION	NULL	VARCHAR2(240)	process description

AZ_PROCESSES_V

A view based on az_processes table. It provides a view of az_processes filtering processes by their process type. The view contains wizard processes having process type equal to the current implementation mode which is a profile option.

View Definition

```
CREATE VIEW AZ_PROCESSES_V
  as SELECT
      ap.item_type,
 ap.process_name,
 ap.context_id,
 ap.parent_id,
 ap.display_order,
 ap.status_code,
 ap.process_type,
 ap.context_type,
 ap.context_name,
 ap.comments
  from az_processes ap
 where ap.process_type = nvl(FND_PROFILE.VALUE_WNPS('AZ_CURRENT_MODE'), 'IMP')
```

Column Descriptions

Name	Null?	Type	Description
ITEM_TYPE	NULL	VARCHAR2(8)	Refer to base table.
PROCESS_NAME	NULL	VARCHAR2(30)	Refer to base table.
CONTEXT_ID	NULL	NUMBER(15)	Refer to base table.
PARENT_ID	NULL	VARCHAR2(240)	Refer to base table.
DISPLAY_ORDER	NULL	NUMBER(15)	Refer to base table.
STATUS_CODE	NULL	VARCHAR2(15)	Refer to base table.
PROCESS_TYPE	NULL	VARCHAR2(15)	Refer to base table.
CONTEXT_TYPE	NULL	VARCHAR2(15)	Refer to base table.
CONTEXT_NAME	NULL	VARCHAR2(80)	Refer to base table.
COMMENTS	NULL	VARCHAR2(2000)	Refer to base table.

AZ_PROCESS_ACTIVITIES_V

AZ_PROCESS_ACTIVITIES_V is a view that stores the definition of a process. Each row includes the item type, process name, context id as well as the complete flag, description and user entered comments of a process. This view is mainly used by the Implementation Wizard form.

View Definition

```
CREATE VIEW AZ_PROCESS_ACTIVITIES_V
  as SELECT
     distinct azp.item_type, azp.process_name, azp.context_id,
     azp.complete_flag, azp.comments, wav.version, wav.description
   from az_processes azp, wf_activities_vl wav
where azp.item_type = wav.item_type
and azp.process_name = wav.name
and wav.end_date is null
```

Column Descriptions

Name	Null?	Type	Description
ITEM_TYPE	NOT NULL	VARCHAR2(30)	Refer to base table.
PROCESS_NAME	NOT NULL	VARCHAR2(30)	Refer to base table.
CONTEXT_ID	NOT NULL	NUMBER(15)	Refer to base table.
COMPLETE_FLAG	NOT NULL	VARCHAR2(1)	Refer to base table.
COMMENTS	NULL	VARCHAR2(2000)	Refer to base table.
VERSION	NOT NULL	NUMBER	Refer to base table.
DESCRIPTION	NULL	VARCHAR2(240)	Refer to base table.

AZ_PRODUCT_FLOWS

AZ_PRODUCT_FLOWS stores information about the runnable workflow processes for each application. Each record contains the unique identifier of the application, the item type, process name, and context type, as well as the parent process group that it is associated with and its display order in the hierarchy tree.

Foreign Keys

Primary Key Table	Primary Key Column	Foreign Key Column
AZ_GROUPS	GROUP_ID PROCESS_TYPE	PARENT_ID PROCESS_TYPE

Column Descriptions

Name	Null?	Type	Description
PROCESS_TYPE	NOT NULL	VARCHAR2(30)	Implementation type for process, Fresh or Upgrade
APPLICATION_ID (PK)	NOT NULL	NUMBER(15)	Application identifier
ITEM_TYPE (PK)	NOT NULL	VARCHAR2(30)	Item type
PROCESS_NAME (PK)	NOT NULL	VARCHAR2(30)	Internal name of the process activity
DISPLAY_ORDER	NOT NULL	NUMBER(15)	Display order of the process in the hierarchy tree
CONTEXT_TYPE	NOT NULL	VARCHAR2(30)	Context type of the process (BG for business group, IO for inventory organization, OU for operating unit, SOB for set of books)
PARENT_ID	NOT NULL	VARCHAR2(240)	Parent process group that the process is associated with

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_PRODUCT_FLOWS_N1	NOT UNIQUE	1 2	PARENT_ID PROCESS_TYPE
AZ_PRODUCT_FLOWS_N2	NOT UNIQUE	1 2	ITEM_TYPE PROCESS_NAME
AZ_PRODUCT_FLOWS_U1	UNIQUE	1 2 3	APPLICATION_ID ITEM_TYPE PROCESS_NAME

AZ_PRODUCT_PHASES

AZ_PRODUCT_PHASES stores information about the phase for each installed product. Each record contains the unique identifier of the application and its corresponding phase.

Column Descriptions

Name	Null?	Type	Description
APPLICATION_ID (PK)	NOT NULL	NUMBER(15)	Application identifier
PHASE	NOT NULL	NUMBER(15)	Phase in the implementation cycle
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	NULL	NUMBER(15)	Standard Who column

Indexes

Index Name	Index Type	Sequence	Column Name
AZ_PRODUCT_PHASES_U1	UNIQUE	5	APPLICATION_ID

AZ_PRODUCT_PHASES_V

AZ_PRODUCT_PHASES_V is a view that stores information about the phase for each installed product defined in AZ_GROUPS. Each row includes the unique identifier and name of an application as well as its corresponding phase. This view is mainly used by the Define Phase form.

View Definition

```
CREATE VIEW AZ_PRODUCT_PHASES_V
  as SELECT
     app.application_id, fav.application_name, fav.application_short_name, app.phase

   from az_product_phases app,
          fnd_application_vl fav,
          fnd_product_installations fpi,
          az_groups azg
where app.application_id = fav.application_id
and fav.application_id = fpi.application_id
and fpi.status = 'I'
and fpi.application_id = azg.application_id
and azg.process_type = 'IMP'
```

Column Descriptions

Name	Null?	Type	Description
APPLICATION_ID	NULL	NUMBER(15)	Refer to base table.
APPLICATION_NAME	NULL	VARCHAR2(240)	Refer to base table.
APPLICATION_SHORT_NAME	NULL	VARCHAR2(50)	Refer to base table.
PHASE	NULL	NUMBER(15)	Refer to base table.

AZ_TASKS_V

AZ_TASKS_V is a view containing all the tasks, or items in Workflow terminology, belong to AZ. Each row includes item key, item type, process name, activity instance id, begin date, end date, assigned user, status, and duration. It is built upon WF_ITEMS, WF_ITEM_ATTRIBUTE_STATUSES, WF_ITEM_ATTRIBUTE_VALUES, and WF_PROCESS_ACTIVITIES.

View Definition

```
    CREATE VIEW AZ_TASKS_V
      as SELECT
         wi.item_type,
    wi.item_key,
    wi.root_activity,
    TO_NUMBER(wiav2.text_value) context_id,
    ap.context_name,
    wi.begin_date,
    wi.end_date,
    SUBSTR(wiav1.text_value, 1, 100) assigned_user,
    wias.process_activity activity_id,
    decode (wias.activity_status, 'ACTIVE', 'A', 'C') status,
    decode(wi.end_date, NULL, round(months_between(sysdate, wi.begin_date)*31),
    round(months_between(wi.end_date, wi.begin_date)*31) ) duration


       from
    wf_item_activity_statuses wias,
    wf_item_attribute_values wiav1,
    wf_item_attribute_values wiav2,
    wf_items wi,
    wf_process_activities wpa,
    az_processes ap
    where
    wi.item_type like 'AZW%'
    and wiav1.item_key = wi.item_key
    and wiav1.item_type = wi.item_type
    and wiav1.name = 'AZW_IA_ROLE'
    and wiav2.item_key = wi.item_key
    and wiav2.item_type = wi.item_type
    and wiav2.name = 'AZW_IA_CTXT_ID'
    and ap.item_type = wi.item_type
    and ap.process_name = wi.root_activity
    and ap.context_id = TO_NUMBER(wiav2.text_value)
    and wias.item_type = wiav1.item_type
    and wias.item_key= wiav1.item_key
    and wias.process_activity = wpa.instance_id
    and wi.root_activity = wpa.activity_name
    and wpa.process_item_type = wi.item_type
    and wpa.process_name = 'ROOT'
```

Column Descriptions

Name	Null?	Type	Description
ITEM_TYPE	NULL	VARCHAR2(8)	Refer to base table.
ITEM_KEY	NULL	VARCHAR2(240)	Refer to base table.
ROOT_ACTIVITY	NULL	VARCHAR2(30)	Refer to base table.
CONTEXT_ID	NULL	NUMBER(15)	Refer to base table.
CONTEXT_NAME	NULL	VARCHAR2(80)	Refer to base table.
BEGIN_DATE	NULL	DATE	Refer to base table.
END_DATE	NULL	DATE	Refer to base table.
ASSIGNED_USER	NULL	VARCHAR2(100)	Refer to base table.
ACTIVITY_ID	NULL	NUMBER	Refer to base table.
STATUS	NULL	VARCHAR2(1)	Refer to base table.
DURATION	NULL	NUMBER	Refer to base table.

Index

Reader's Comment Form

Oracle Applications Implementation Wizard Technical Reference Manual
A80906–01

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.

CAUTION!

This document contains proprietary, confidential information that is the exclusive property of Oracle Corporation.

If you are an Oracle customer, this numbered copy is registered in your site's name. Use, duplication, and disclosure are severely restricted and subject to restrictions stated in your contract with Oracle Corporation.

If you are an Oracle employee, this numbered copy is registered in your name and must be returned if you leave the company. It is your responsibility not to disclose this information to anyone outside of Oracle who is not licensed to use this product. If you have a prospect who needs to see this information, contact mailid APPSAPP to obtain a special nondisclosure agreement.

If you are neither an Oracle customer with a valid contract for the use of this document nor an Oracle employee, then you received this document in an unauthorized manner and are not legally entitled to possess or read it. Please return it immediately to Oracle. You may call collect to the office of the Executive Vice President Applications Division, to arrange its return.

Copy #

ORACLE®

A80906-01