Oracle® Advanced Product Catalog

Implementation Guide

Release 11*i*

Part No. B16452-02

September 2005

ORACLE

Oracle Advanced Product Catalog Implementation Guide, Release 11*i*

Part No. B16452-02

Primary Author: Moitrayee Bhaduri

Contributing Author: James Aliferis, Jayanta Mukhopadhyay, Sachin Patel, Ravi Polisetty, Alison Schofield, Kunal Desai, Anu Mandalam, Sandeep Sood

Contents

4   Defining Catalogs

5   Defining Change Categories and Types

6   Defining Item Structures

7 Setting Up Document Management

8 Defining Roles

9 Implementing Role Based Security

10 Understanding Product Data Synchronization

A Change Management Workflows

B Reading Data for User-Defined Attributes

C Synchronizing Item User-Defined Attributes with Item Descriptive Elements

Send Us Your Comments

Oracle Advanced Product Catalog Implementation Guide, Release 11*i*

Part No. B16452-02

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used 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?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appsdoc_us@oracle.com
- FAX: 650-506-7200 Attn: Oracle Supply Chain Management Documentation Manager
- Postal service:
  Oracle Supply Chain Management Documentation Manager
  Oracle Corporation
  500 Oracle Parkway
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Intended Audience

Welcome to Release 11*i* of the *Oracle Advanced Product Catalog Implementation Guide*.

Implementers

See Related Documents on page x for more Oracle Applications product information.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/ .

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

1 Implementation Overview

This section contains an overview of the steps you need to complete to set up Oracle Advanced Product Catalog (APC).

Related Documents

Oracle Advanced Product Catalog User's Guide

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using an Oracle Applications form can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

1

Implementation Overview

This section contains an overview of the steps you need to complete to set up Oracle Advanced Product Catalog (APC).

This chapter covers the following topics:

- Oracle Applications Implementation Wizard
- Set Up Oracle Applications Technology
- Setup Prerequisites
- Using Oracle iSetup
- Setup Steps
- Profile Options

Oracle Applications Implementation Wizard

If you are implementing more than one Oracle Applications product, you may want to use the Oracle Applications Implementation Wizard to coordinate your setup activities. The Implementation Wizard guides you through the setup steps for the applications you have installed, suggesting a logical sequence that satisfies cross–product implementation dependencies, and reduces redundant setup steps. The Wizard also identifies steps that can be completed independently, by several teams working in parallel, to help you manage your implementation process most efficiently. You can use the Implementation Wizard as a resource center to see a graphical overview of setup steps, read online help for a setup activity, and open the appropriate setup window. You can also document your implementation, for further reference and review, by using the Wizard to record comments for each step.

See Also: *Oracle Applications Implementation Wizard User's Guide* and *Oracle Applications System Administrator's Guide*.

Set Up Oracle Applications Technology

This overview explains how to implement the parts of Oracle Applications specific to Oracle Advanced Product Catalog. The Implementation Wizard guides you through the entire Oracle Applications setup, including system administration. However, if you do not use the Wizard, you need to complete several other setup steps, including:

- Perform system wide setup tasks such as configuring concurrent managers and printers

- Manage function and data security, which includes setting up responsibilities to allow access to a specific set of business data and complete a specific set of transactions, and assigning individual users to one or more of these responsibilities.

Also, if your product uses Oracle Workflow to manage the approval of change requests (for example) or to send change notifications, you need to set up Oracle Workflow.

See Also:

Oracle Applications Implementation Wizard User's Guide

Oracle Applications System Administrator's Guide

Oracle Workflow Guide

Setup Prerequisites

Before setting up Oracle Advanced Product Catalog, you must complete the *required* setup steps for the following:

- In Oracle Inventory, see "Setting Up" in the *Oracle Inventory User's Guide*

- In Oracle Bills of Material, see "Setting Up" in the *Oracle Bills of Material User's Guide*

- In Oracle Engineering, see "Setting Up" in the *Oracle Engineering User's Guide*

- In Oracle Projects, if you intend on using lifecycle tracking projects, see "Setup and Implementation" in the *Oracle Projects Documentation Update Notes Family Pack L.*

- In Oracle Sourcing, if you intend to collaborate with suppliers complete the setup steps for defining suppliers see the *Oracle Sourcing Implementation and Administration Guide*

- In Oracle Receivables, if you intend to collaborate with customers complete the setup steps for defining customers see "Setting Up Receivables" in the *Oracle Receivables User's Guide*

Using Oracle iSetup

If you are implementing a new instance of the Oracle e-Business Suite and have relatively standard implementation requirements, you can use Oracle iSetup to rapidly configure the following applications:

- Oracle Purchasing

- Oracle Inventory

- Oracle Bills of Material

- Oracle Order Management

- Oracle Shipping

- Oracle Costing

- Oracle Planning

- Oracle Engineering

- Oracle Work in Process

- Oracle General Ledger

- Oracle Payables

- Oracle Receivables

- Oracle Cash Management

- Oracle Assets

Oracle iSetup simplifies the setup process by providing templates and predefined setup steps that may or may not apply to your enterprise.

Setup Steps

Following is a summary of the setup steps for Oracle Advanced Product Catalog:

- Defining Responsibilities and Users

- Defining the Item Catalog

- Defining Catalogs

- Defining Change Categories and Types

- Defining Item Structures

- Setting Up Document Management

- Defining Roles

- Implementing Role-Based Security

Profile Options

Oracle Advanced Product Catalog provides several system profile options to enable or change the behavior of certain features in the application. Following is a description of each of the Oracle Advanced Product Catalog system profiles and their default values.

APC Profile	Description	Default Value
EGO: Enable exporting to Excel	Governs the action of exporting search results from APC. **Note:** You should install Web ADI (see "About Oracle Product Lifecycle Management (PLM) Family Pack C" Metalink note 282980.1) before changing the value to Yes. Changing the value to Yes enables users to export certain search results data from APC.	No

EGO: Internal User Default Role for Items	Governs the default role that is set up for internal users who access items. This role is assigned to internal users for all items in the organizations to which the users have access.	Design Reviewer
	Design Reviewer is seeded for Forms interoperability.	
	Note: Forms does not have data security, so users must be given additional privileges on specific items (either directly or via inheritance) to perform actions on items in APC.	
	Users with the Design Reviewer role can view/access all item information. If you want to restrict access further, define a custom role and then reset the default role for all enterprise internal users.	
	It is recommended that you always specify a role that minimally has the View Item privilege.	
EGO: Maximum number of rows for sorting	Specifies the maximum number of rows to sort during an item search. If a search retrieves more than the maximum number of rows, the search results are displayed without sorting (a warning is displayed).	2000
	Note: Specifying a value greater than 2000 may have a negative impact on search performance.	
EGO: Number of recent links to display	Specifies the number of links to display in the Recently Visited list	20

EGO: Restrict actions in Item form	Governs the Forms actions for items after APC is installed.	No
	Set to Yes to ensure that APC data level role-based security is honored in Forms. When set to Yes users will be unable to access the following APC functionality in Forms:	
	• Categories	
	• Catalog	
	• Revisions	
	• Organization Assignments	
	• Attachments (disables attachments)	
	Keep the default value No if you wish to make all actions available in Forms.	

ENG: Internal User Default Role for Changes	Governs the default role that is set up for internal users who access change objects. This role is assigned to internal users for all change objects in the organizations to which the users have access.	Reviewer
	Reviewer is seeded for Forms interoperability.	
	Note: Forms does not have data security, so users must be given additional privileges on specific change objects (either directly or via inheritance) to perform actions on change objects in APC.	
	Users with the Reviewer role can view/access all information for a change object. If you want to restrict access further, define a custom role and then set reset the default role for all enterprise internal users.	
	It is recommended that you always specify a role that minimally has the View Item privilege.	
EGO GDSN ENABLED	Governs the GDSN Synchronization functionality.	No
	Changing the value to Yes enables GDSN functionality.	
	Once you turn the flag to yes, then	
	• GDSN default functional area appears	
	• GDSN attributes appear in setup list page	
	• GDSN attributes get associated to Item Catalog Category	
	• GDSN Syndicated functional area flag on the Create Item and Functional Classification page gets enabled.	

2

Defining Responsibilities and Users

Defining Responsibilities and Users

Oracle APC enables you to securely collaborate with internal users, suppliers, and customers. You can set up users and grant them responsibilities to enable specific business functions. You can further restrict access to specific items, catalogs, or change management objects with role-based security.

Task	Required?
Creating Responsibilities	
Creating Internal Users	Yes
Setting Up Supplier Users	
Setting Up Customer Users	

Creating Responsibilities

The top-level menu for the APC-seeded responsibilities (Development Manager, Development Engineer, and Supplier Engineer) is the EGO Manager Menu (EGO_MANAGER_MENU); you should view the APC-seeded menus and functions before creating ones of your own. To view the APC-seeded menus and functions, open the EGO Manager Menu in the Application Developer's Menu form and click the View Tree button. In the Menu Viewer you can expand each menu as you decide which menus to include in your custom responsibilities. If a node appears with no prompt under a menu, then you should include that function to ensure that all the menu items work correctly. You can use menu exclusion rules when defining your responsibilities if you need to exclude any specific menus or functions for a user.

See Also: For more details about creating responsibilities, see the *Oracle Applications System Administrator's Guide*.

Oracle APC top level menu with seeded responsibilities

Menu Viewer

Defining responsibilities with menu exclusions

Creating Internal Users

You (the system administrator) can define E-Business Suite users and assign them responsibilities that grant them access to specific E-Business Suite applications such as Oracle Advanced Product Catalog. After creating an employee, customer or supplier, create a user account to which you then link the employee, customer, or supplier. Each internal user should have an employee ID with an e-mail address specified so he or she can receive notifications. You can assign internal users a custom responsibility or one of the following seeded responsibilities:

- Development Engineer
- Development Manager
- Manufacturing Engineer
- Product Manager
- Program Manager
- Project Engineer
- Supplier Engineer

See Also: For more details about creating internal users, see the *Oracle Applications System Administrator's Guide*.

Setting Up Supplier Users

Oracle APC enables you to securely collaborate with suppliers. You can set up E-Business Suite users for each of the suppliers with whom you collaborate on product and component designs.

Assuming the supplier company has already been defined in Oracle Purchasing or Oracle Sourcing, you can register a supplier user in Oracle Sourcing using the Sourcing Super User responsibility. You need to specify the supplier's e-mail address as the user ID so that the system can e-mail the supplier his or her password. After the supplier user is registered via Oracle Sourcing, you can assign him or her a responsibility such as Supplier Engineer, which specifies the APC functions available to that supplier user. For example, the seeded responsibility Supplier Engineer provides a restricted set of APC functions that enable the supplier to view specific items (secured with item roles), create issues and change requests for an item, but not create or view change orders.

See Also: For more details about registering and setting up supplier users, see the *Oracle Sourcing Implementation and Administration Guide* and *Oracle Applications System Administrator's Guide*.

Setting Up Customer Users

Oracle APC enables you to securely collaborate with your customers. You can set up E-Business Suite users for each of the customers with whom you collaborate on product and component information.

Assuming that you have already defined the customer company in the Oracle E-Business Suite, create customer users by selecting the Trading Community Manager responsibility. In the Customers – Standard form create a new Customer Type of Person. You must specify a valid Address.

Then navigate to the Party Relations window and assign the person to the customer company as follows:

Relationship Type

EMPLOYMENT

Relation

Employee Of

Object

Company name

Start Date

Specify a start date. After assigning a person to the customer company, select the System Administrator responsibility to define the customer user. Then, in the Users form, enter the required information and specify the customer person name in the Customer field.

See Also: For more details about setting up customer users, see the *Oracle Receivables Users Guide* and *Oracle Applications System Administrator's Guide*.

Creating a new customer person

Setting up party relations for a new customer person

Defining a customer user

3

Defining the Item Catalog

Overview of Item Catalogs

Oracle APC enables you to define hierarchical taxonomies of products, components, and items. The Item Catalog contains all the items in the system; other catalogs are a sub-set of the item catalog. For example the Product Catalog contains all products that are sold. Customers can also define additional catalogs (see Step 3: Defining Catalogs) for browsing and reporting purposes. Each catalog has a hierarchy of categories containing assigned items. The item catalog and product catalog are pre-defined catalogs.

Task	Required?
Build the Item Catalog Text Index	Yes
Defining Item Catalog Categories	Yes
Defining Value Sets for User-Defined Attributes	
Defining Item Attributes and Attribute Groups	
Setting Up Item Attribute Group Security	
Implementing User-Defined Functions for Item Attributes	
Associating Item Catalog Attributes	
Defining Item Attachment Categories	
Associating Item Attachment Categories	
Defining Lifecycle Phases	
Defining Lifecycles	
Associating Lifecycles to Item Catalog Categories	
Defining Item Catalog Criteria Templates	
Defining Item Catalog Results Formats	
Defining Item Catalog Import Formats	

Build the Item Catalog Text Index

To take advantage of the Oracle Text search features available with Oracle APC (such as "keyword search", "stemming search", and "fuzzy search") run the concurrent program 'Item Catalog Text Index Build'. Any user with the Development Manager responsibility can submit this concurrent program from the Submit Request form. Select the concurrent request parameter Action=CREATE when submitting the request.

Defining Item Catalog Categories

The *item catalog* contains all items defined for your company or organization. Within the item catalog, *item catalog categories* provide a way to logically classify items that have similar characteristics. So, you can classify a computer monitor in an item catalog category named Computer Parts and Components. Item catalogs are hierarchical, and relationships between the various item catalog categories (such as an item's catalog category attributes) are inherited. You can define a rich hierarchical taxonomy of item catalog categories that enable you to set up item attribution, lifecycles and their change policies, security, attachment categories, results formats, criteria templates, and import formats.

When defining item catalog categories, do the following:

- Create a list of all your items.

- Classify the items in unique item catalog categories and sub-categories that suit your business needs.

- Consider which category is a sub-classification within a category (for example, SRAM, DRAM could be subcategories of the item catalog category Memory).

- List the required and optional item attributes for each item catalog category

Following are some of the basic Advanced Product Catalog item catalog category terms you will encounter throughout this guide:

- **Item Catalog Categories**

  Item catalog categories inherit attributes from the parent category; so consider which attributes to associate at higher levels in the item catalog category. Inheritance helps to simplify maintenance.

- **Item Lifecycles**

  Item lifecycles enable you to track and control the lifecycle phases of items. You can specify an item lifecycle for each item catalog category. Item lifecycles are inherited from the parent item catalog category.

- **Change Policy**

  Item behavior is described by defining a change policy for each lifecycle phase in a given lifecycle.

- **Security**

  You can control item creation, viewing and update access by assigning a role on the item catalog category to a user or group of users.

- **Attachment Categories**

You can classify a document that is attached to an item using an attachment category. Attachment categories are inherited from the parent catalog category, and those most commonly used can be defined at a higher node. You can then add more specific attachment categories for the child item catalog categories.

> **Note:** Catalog Groups defined in Oracle Inventory will appear as Item Catalog Categories in Oracle Advanced Product Catalog.

> **See Also:** For details about seeded base attributes, see the *Oracle Inventory User's Guide*.

Defining Value Sets for User-Defined Attributes

User-defined attributes capture all the detailed information (such as cost information) about an object (items, change requests or change orders). The item catalog enables you to create user-defined attributes with validation logic to support the needs of your organization. To do so, create value sets and associate the value sets with user-defined attributes. Attributes can have a static or dynamic list of valid values, or a range of values.

Prior to creating value sets for user-defined attributes establish the validation rules for each attribute as follows:

- Choose the data type for the values.

- Know the range of the values.

- Know which set of actual values will satisfy the required condition for the attribute. For static values, specify a fixed set of values. Dynamic values are retrieved from a database table or view.

- Select the manner in which the values will be displayed: list of values (LOV), pop list (dropdown) or radio group.

The following table shows the validation rules for the attribute group Benchmark Rating.

Attribute Group	Attributes	Data Type	Value Range	Values	Display	Create Value Sets
Benchmark Rating	Risk Rating	Number	1 to 5	Static: 1,2,3, 4,5	Pop list	Rating
Benchmark Rating	Quality Rating	Number	1 to 5	Static: 1,2,3, 4,5	Pop list	Rating
Benchmark Rating	I/O Subsystem Rating	Number	1 to 5	Static: 1,2,3, 4,5	Pop list	Rating
Benchmark Rating	Video System	Number	1 to 5	Static: 1,2,3, 4,5	Pop list	Rating
Benchmark Rating	Overall Rating	Number	1 to 5	Static: 1,2,3, 4,5	Pop list	Rating
Benchmark Rating	Failure Rate	Number	Min 0 Max100	Within the range	Text Field	Percent Range
Benchmark Rating	Supplier	Char	N/A	Dynamic: Queried from tables in the database	LOV	Primary Supplier

After analyzing the required validation rules, create the value sets as follows:

1. Specify the Name and Description.

> **Important:** Once specified, you cannot edit the value set name.

2. Select the Data Type. The data type determines the values that are available in the value set. An attribute's data type must match the data type specified for that attribute's value set. In the preceding table, the data type Number has been specified for the value set Rating because it contains all numeric values. In comparison, the value set Primary Supplier uses the data type Char because it only contains text values.

> **Important:** You cannot edit the data type once the value set has been created.

3. Choose the value for Maximum Size if you wish to limit the user's input in the attribute text field. For example, in some cases you may wish to limit the number of characters in the attribute Date to 10 characters, or the number of characters in Name to 50. Keep the default value of 0 if you wish to omit this particular validation.

4. Select the Validation Type:

- **None**: Select None to indicate that there is no explicit set of values against which the user's input is validated.

- **Independent** : You define the explicit values against which the user's input is validated here. To create explicit values, select Independent and click Add

Values. Enter each value by specifying its internal/display name, sequence and the start/end date.

For example, the value set Rating can have certain values disabled by setting the end date.

You can specify the way the values are displayed as follows:

- *List of values* : Users select values by clicking the list of values (LOV).

- *Pop list* Users select values from a dropdown.

> **Note:** Display as Radio Group can be set while creating or editing attributes. For more details see Definint Item Attributs and Attribute Groups. Oracle recommends that when you use the value set type Independent, you save it as a pop list.

- *Table*: The explicit values against which the user's input is validated comes from a database table. To create explicit values, select Independent and click Add Values, or click Table and then click Edit Table Information. This information can be dynamically changed.

> **Note:** Value sets using the Table validation type are always displayed as a list of values (LOV).

Example of the value set Primary Supplier

The **Value Set Details** page is displayed upon completion. You can edit certain settings after creation by clicking Update. You can find all existing value sets on the **Value Sets**

page. You can search for value sets using the criteria Name, Description, Data type, or Validation type.

Value sets described in the attribute group Benchmark Rating

Maintain Value Sets

| Search Name | | * Data Type | Number | |
| Description | | * Validation Type | All | |

(Go)

Value Sets
From here, you can search for existing Value Sets or create new Value Sets.

(Create)

Name	Description	Data Type	Maximum Size	Validation Type	Minimum Value	Maximum Value
Percent_Range	Value Set for Percentage Range Validation	Number	0	None	0	100
Rating		Number	0	Independent		

You can use one of the seeded value sets (Yes or No) that are automatically displayed when you select the Display Option for the attribute. You can see some of the seeded value sets on the **Value Set** page; these are used to support images and attachments in the long description on the **Item Summary** pages.

See Also: For more details about creating value sets see the APC online Help (iHelp).

Defining Item Attributes and Attribute Groups

Every catalog item has a set of operational attributes that determines the behavior of features in other E-Business Suite applications. You can define attributes that more specifically identify items, their characteristics and specifications, and capture business process information. The item catalog enables you to create user-defined attributes with certain validation logic (for example, value sets) and indexes.

Attributes are defined by their names and values, and are saved within attribute groups. You can associate attribute groups with an item catalog category; consequently, items belonging to the same catalog category have the same attributes. You can reuse the same attribute group across different item catalog categories. You can also set up as many attributes as necessary to define an item. Later, users can enter the values for the attributes on the item pages. Although item categories inherit attributes from the parent category, each child category can also have its own unique attributes. You can also configure how attributes are displayed to improve usability. You can specify attributes at the item or item revision level.

See Also: For details about associating attributes, see Associating Item Catalog Attributes.

Prior to creating user-defined attributes you should:

- Group related attributes within the same attribute group. The following table shows some examples of attribute groups.

Attribute Group	Attributes	Data Type
Capacitor Specifications	Dielectric	Char
	Voltage	Number
	Tolerance	Number
	Minimum Temperature	Number
	Maximum Temperature	Number
Capacitor Packaging Specs	Package Type	Char
	Size	Number
	Case Size	Number

- Create an object role (item) with custom privileges to define attribute group security (if necessary). You can secure the attribute group by setting Edit/View privileges. Later on, only users with certain roles can view or edit those attributes. **See Also:** For more details, see "Implementing Item Attribute Group Security."

- Determine the data type of the attributes (for example, number, char, date).

  Important: You cannot edit the data type after you save the attribute.

- Set up the Unit of Measure Classes (for example, Currency) and Units of Measure (for example, US Dollars).

- Establish the validation rules for each attribute, if necessary. For more details, see Defining Value Sets for User Defined Attributes.

- Decide whether you want the attribute to be searchable. If you want the attribute to be searchable, then (when creating the attribute) select an indexed column in the database. Ensure that the Indexed checkbox is selected before you complete the attribute definition. If no indexed columns are available in the database, and you still want a searchable attribute, select a non-indexed column, and ensure the Indexed checkbox is selected before completing the attribute definition; then the database column is automatically indexed. If you do not want the attribute to be searchable, and the only columns available in the database are indexed, then ensure that the Indexed checkbox is not selected before completing the attribute definition.

- Choose the display format of the attribute group:

  - **Single row** Displays a Text field with a value.

  - **Multi-row** Displays multiple values (rows) for the same attributes (column) in a table.

    Important: Define which attributes or combination of attributes will maintain uniqueness of records in cases where the attribute group is displayed as multi-row.

The figure on "Defining item attributes and attribute groups" shows how the display options are set and used when specifying the value for attributes. A few attributes (such as Risk Rating, and Quality Rating) are set to display as a text field. However, the value

set Rating is saved as a pop list; therefore, the attribute values are selected from the dropdown. The attribute Supplier uses the Primary Supplier value set, which is a list of values. The attribute Mean Time Between Failure uses the data type Number. Failure Rate also uses the Number data type and the Unit of Measure (UOM) is Percent.

Defining item attributes and attribute groups

Attribute Group Details

Update

Internal Name Benchmark_Ratings
Display Name Benchmark Ratings
Description Benchmark Ratings
Multi-Row No

Data Security

View Privilege
Edit Privilege

Attributes

Add Attribute

Select Item(s) and ...   Delete

Select All | Select None   Previous Next 10

Select	Sequence	Internal Name	Display Name	Data Type	Display As	Value Set Name	Enabled	Required	Indexed	Column	Edit
☐	10	Risk_Rating	Risk Rating	Number	Text Field	Rating	Yes	No	No	N_EXT_ATTR1	✐
☐	20	Quality_Rating	Quality Rating	Number	Text Field	Rating	Yes	No	No	N_EXT_ATTR2	✐
☐	30	CPU_Rating	Atomic CPU Rating	Number	Text Field	Rating	Yes	No	No	N_EXT_ATTR3	✐
☐	40	Video_Subsystem	Video Subsystem	Number	Text Field	Rating	Yes	No	No	N_EXT_ATTR4	✐
☐	60	IO_Subsystem_Rating	I/O Subsystem Rating	Number	Text Field	Rating	Yes	No	No	N_EXT_ATTR5	✐
☐	70	Overall_Rating	Overall Rating	Number	Text Field		Yes	No	No	N_EXT_ATTR6	✐
☐	80	Supplier	Supplier	Char	Text Field	PrimarySupplier	Yes	No	No	C_EXT_ATTR1	✐
☐	90	MTBF	Mean Time Between Failure (Obsolete)	Number	Text Field		No	No	No	N_EXT_ATTR7	✐
☐	91	MTBF1	Mean Time Between Failure	Number	Text Field		Yes	No	No	N_EXT_ATTR10	✐
☐	100	Failure_Rate	Failure Rate (Obsolete)	Number	Text Field	Percent_Range	No	No	No	N_EXT_ATTR9	✐

Also notice in the figure above that certain attributes are set with the Enabled parameter set to No, indicating that such attributes are not displayed. You can always disable certain attributes. However, you cannot delete attributes after an attribute group has already been associated with an item catalog category.

Defining item attributes and attribute groups (setting attributes values)

Benchmark Ratings

Calculate Overall Rating

▪ Indicates required field

Risk Rating 4
Atomic CPU Rating 2
I/O Subsystem Rating 3
Supplier Advantage Corp
Mean Time Between Failure 2000 Day
Failure Rate 0.03 %

Quality Rating 5
Video Subsystem 3
Overall Rating 3

Cancel Apply

The following figure is an example of the attribute group Marketing. Notice that the attribute Name is maintained as a Part of Unique Key and is set as required.

You can add/edit the unique key as long as doing so does not destroy the uniqueness (creating duplicates) of existing records.

Defining item attributes and attribute groups

See Also: For more details about creating item attribute groups and attributes, see the APC online Help (iHelp).

Display Attachments on Item Attribute Pages

You can display items or change attachments as attributes on an item attribute page. Select a specific file from the existing item/change attachment list to display the file as a link within an attribute group on an item or change object page. You can then set up a Results Format that includes the document attribute as a column. Consequently, the link to the attachment is displayed in the search results.

To display an attachment on an item attribute page:

1. Create a value set that queries the values for the object primary key and retrieves the information for the object attachment data.

2. Create an attribute group with an attribute where Display As is set to Attachment.

3. Associate the value set to the attribute.

4. Associate the attribute group to an item catalog category.

5. Create an item page to display the attribute group.

6. Add document attachments to the item attachment list.

See Also: For more details, see:

- Defining Item Attributes and Attribute Group

- Define Value Set for Attribute

- Associating Item Catalog Attributes

> **Note:** To see Attachments in the Display As dropdown, your attribute should use the Number data type.

When creating a value set to display item attachments, select the Table validation type and the Number data type (as shown in the following table).

Attribute Group: *Computer Specifications*	**Attribute Group:** *Computer Specifications*	**Attribute Group:** *Computer Specifications*	**Attribute Group:** *Computer Specifications*
Attribute: *Release Notes*	**Attribute**: *Release Notes*	**Value Set:** *Item Attachment*	**Value Set:** *Item Attachment*
Data Type	**Display as**	**Data Type**	**Validation Type**
Number	Attachment	Number	Table

Create a value set for an attribute to display an item attachment

Associate value set to the attribute

Select attachments to be displayed on attribute page

Item attribute page displays link to one of the attached files

Setting Up Item Attribute Group Security

When implementing role-based item security you can set up privileges to control the view and edit permissions for specific item attribute groups. You can control which users can view and/or edit certain attribute groups for an item by assigning a role granting those specific privileges. By default, an item role's View Item and Edit Item privileges control whether or not you can view or edit item attributes that are not controlled specifically at the item attribute group level. In other words, when implementing item security you do not have to specify a view or edit privilege for each item attribute group.

Example: Attribute Group Security

Suppose your company is designing, along with your supplier, a new motherboard for its next generation of desktop computers. To improve design collaboration you would like to securely share item information about the motherboard—both internally between departments and externally with your suppliers and contract manufacturers. The Supplier Engineer should only be able to view specific item attribute groups such as the Technical Specifications. The Supplier Engineer should not be able to view the Market Research attributes. The Engineering Manager and Marketing Manager should be able to view and edit the Market Research attributes, while a Design Engineer should only view the Market Research attributes. There are three sets of Market Research attributes (attribute groups): Key Metrics, Target Markets, and Competitors.

To implement attribute group security for this scenario:

1. Select the Application Developer responsibility, navigate to the Form Functions form, and create Form Functions for each privilege that controls view and edit permissions for the Market Research attribute groups.

2. Select the Development Manager responsibility and navigate to the Setup Workbench. In the **Attribute Group Details** page for each Market Research-related attribute group (for example, Target Markets) specify the View Privilege (for example, View Target Markets) and Edit Privilege (for example, Edit Target Markets) in the Data Security section.

3. On the **Item Role Detail** page for the Marketing Manager and Engineering Manager roles, grant the following privileges:

   • View Target Markets

      • View Key Metrics

      • View Key Metrics

      • View Competitors

      • Edit Target Markets

      • Edit Key Metrics

      • Edit Competitors

         For the Design Engineer role, grant the following privileges:

      • View Target Markets

      • View Key Metrics

      • View Competitors

   Do not grant any of the Market Research privileges to the Supplier Engineer role.

   See Also: For more details about setting up item attribute group security, see the following topics in the APC online Help (iHelp):

   • Creating Custom Privileges

   • User Defined Attributes

   • Managing Roles

Defining Form Functions to create user-defined item role privileges

Form Functions — Region tab

Function	Object	Region Application	Region Code
EGO_EDIT_TARGET_M	Item		
EGO_VIEW_TARGET_M	Item		
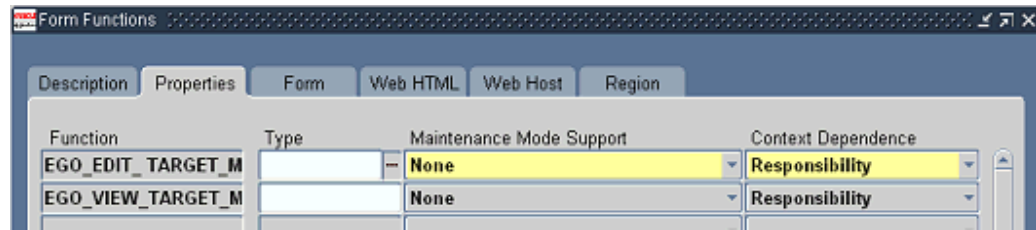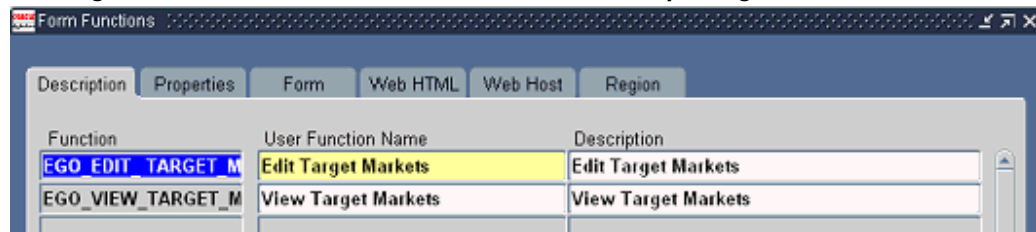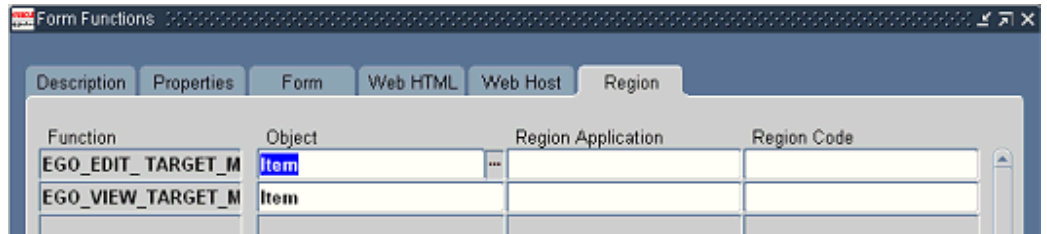
Defining Item Attribute group with view and edit privileges

Attribute Group Details

Update

Internal Name Target_Markets
Display Name Target Markets
Description
Multi-Row Yes

Data Security

View Privilege View Target Markets
Edit Privilege Edit Target Markets

Attributes

Maintain Unique Key   Add Attribute Groups

Select Item(s) and ...

Delete

Select All | Select None

Select	Sequence	Internal Name	Display Name	Data Type	Display As	Value Set Name	Part of Unique Key	Enabled	Required	Indexed	Column	Edit
☐	10	Target_Mkt_Name	Name	Char	Text Field		Yes	Yes	No	Yes	C_EXT_ATTR1	✎
☐	20	Target_Mkt_Description	Description	Char	Text Field		No	Yes	No	No	C_EXT_ATTR2	✎

Associations

Classification	Data Level	Enabled	Display Pages
Computer Parts and Components	Item	Yes	Market Research
Computer Systems	Item	Yes	Market Research

Defining the Marketing Manager role with specific item attribute group privileges

Item Role Detail

Name Marketing Manager
Description Marketing Manager
Object Name Item

Item Privileges

Name	Description
View Target Markets	View Target Markets
Add/Delete Item Attachment	Add/Delete Item Attachment
Edit Competitors	Edit Competitors
Edit Item	Edit Item
Edit Key Metrics	Edit Key Metrics
Edit Target Markets	Edit Target Markets
Grant/Revoke Role on the Item	Grant/Revoke Role on the Item

Defining the Supplier Engineer role with no specific item attribute group privileges

Item Role Detail

(Edit)

Name **Supplier Engineer**
Description **Supplier Engineer**
Object Name **Item**

Item Privileges

Name	Description
View Item Revision List	View Item Revision List
View Item	View Item

Role Mappings

(Edit)

Type	Role Name
Issue	Change Supplier Engineer

Implementing User-Defined Functions for Item Attributes

You can define your own custom logic by adding user-defined functions and actions to existing item pages. Doing so saves you from having to customize an entire page.

Using different algorithms, you can calculate values by passing attribute values to functions. Actions–which can be displayed as buttons or links–become the trigger points for functions. You can determine the conditional visibility of the button or links, and also prompt users based on their input. You can also trigger a function when you save changes to attributes (for details, see "Appendix D: Synchronizing Item User-Defined Attributes with Item Descriptive Elements").

User-defined functions can be Java, URL, or PL/SQL functions. Prior to setting up user-defined functions and actions you should:

To set up user defined attributes:

1. Set up user-defined item attributes. Create the necessary item pages associated with the item catalog categories.

2. Determine which attributes are functions of other attributes. Create user-defined functions to perform the necessary calculations (for example, summation of costs, efforts, ratings).

3. Create custom privileges and roles if you need to secure access and control who can/cannot execute the function/action.

4. Determine the conditions for displaying or changing the prompt of the button/link that executes the function. For example, the button may not be displayed until certain required attributes have values entered. The name of the button may change depending on the values of certain attributes.

   Java Function: Place the Java class files in any directory and append this directory to the Apache servlet classpath.

   PL/SQL Function: Run the PL/SQL package in your custom schema and then create a synonym for this package in the APPS schema.

URL Function: Specify an absolute or relative URL.

See Also: For details about setting up user-defined functions, see "Setting Up Functions" in the APC iHelp.

When setting up parameters for functions:

1. Select the parameter type for each parameter based on whether the corresponding attribute is providing an input parameter to the function or expecting a return value from the function (for example, input, output, input/output).

2. Specify the order sequence of the function parameters; this is the order in which the parameters are passed to a function or procedure. For example, you need to calculate time duration in days by using the attribute group "Duration in days" where Duration = End Date – Start Date.

5.

Sequence	**Attribute Group:** *Duration in Days*	**Attribute Group:** *Duration in Days*	Mapping Attribute & Parameters	**Java Function**: *Duration*	**Java Function**: *Duration*	**Java Function**: *Duration*
Sequence	Attribute Name	Data Type	---------@--------	Parameter Name	Data Type	Parameter Type
1	Start Date	Standard Date	---------@--------	Date 1	Date	Input
2	End Date	Standard Date	---------@--------	Date 2	Date	Input
3	Duration	Number	---------@--------	Result	Integer	Return Value

Once the function and parameters for Duration (based on the Java method that resides in the Java class) are set up, create an action to trigger the function on the item pages. Navigate to the attribute groups list for the corresponding item catalog category (such as Computer Systems) and select the Update action.

See Also: For details about adding actions, see "Adding Actions to an Attribute Group" in the APC iHelp.

When setting up an action:

1. Add the Dynamic Visibility function (written in Java only) to the action if there is a certain condition that must be satisfied to view the action button or link. For example, if one or all of the fields are empty, then the user won't see the button/link to execute the action.

2. Add the Dynamic Prompt function (written in Java only) to the action if the label for the button or link must be changed depending on a certain condition. For example, if the Total Implement Cost field is empty, then the button label is Apply; otherwise the button label is Apply Changes.

3. After creating the action, map the function parameters to the corresponding attributes.

   The mapping section on the **Action Detail**page contains mapping information for function parameters. You can also map function parameters to the object's

primary key value (for example, ORGANIZATION_ID, INVENTORY_ITEM_ID for item objects).

> **Important:** Dynamic Prompt and Dynamic Visibility function parameters must be mapped to the corresponding attributes as well.

6. The following figures show the setup details for the Calculate Grade function. The function was created to calculate the overall rating for items belonging to the Motherboard item catalog category. Benchmark Rating attribute values were used as parameters.

See Also: For details about setting up item attributes see "Defining Item Attributes and Attribute Groups" and "Associating Item Catalog Attributes."

Defining a function

Action associated to attribute group to trigger the function

Action Details for Calculate_Rating

Basic Information

Object Name	Item
Classification	728
Attribute Group	Benchmark Ratings
Sequence	10
Action Name	Calculate_Rating
Description	
Function	Calculate Grade
Security Privilege	
Execution Method	User Action
Display Style	Button
Prompt Application	Oracle Engineering Online
Prompt Message Name	Calculate Overall Rating
Visible	Yes
Dynamic Prompt Function	
Dynamic Visibility Function	

Mappings for Function Calculate Grade

Parameter Name	Mapped To	Mapped Attribute
Component Risk	Attribute Group	Risk Rating
Lead Time	Attribute Group	Quality Rating
Cost	Attribute Group	Atomic CPU Rating
Supplier Risk	Attribute Group	I/O Subsystem Rating

Item attribute page displays action button

Benchmark Ratings

Calculate Overall Rating

* Indicates required field

Risk Rating	4
Atomic CPU Rating	2
I/O Subsystem Rating	3
Supplier	Advantage Corp
Mean Time Between Failure	2000 Day
Failure Rate	0.03 %

Quality Rating	5
Video Subsystem	3
Overall Rating	3 Result

Cancel   Apply

> **See Also:** For more details about user-defined functions, see the APC online Help (iHelp).

Associating Item Catalog Attributes

To display user-defined item attributes, associate the corresponding attribute groups with an item catalog category and specify the item page upon which the attribute group should appear.

Note: You can re-use the same attribute group throughout the item catalog for different item catalog categories. Children categories inherit attribute groups from the parent category. You cannot edit inherited data.

Consider which item catalog categories in the item catalog hierarchy need to display the attribute groups.

Associate an attribute group to a parent category only if you need to view those attributes in all items within that hierarchy. For example, the Benchmark Rating attribute group is associated with the Computer System item catalog category. The Computer System category is the parent category for Desktop and Laptop. All items belonging to the parent category Computer System, as well as to its children Desktop and Laptop, will have Benchmark Rating attributes displayed on the items pages.

To associate an attribute group to an item catalog category:

1. Add the attribute group to the Attribute Groups list for that item catalog category. You can associate attribute groups to an item or item revision.

    > **Note:** Item level attributes are specific only to the Master Organization. However, the item revision level attributes are always organization-specific. For example, the Benchmark Rating attribute group is an item level attribute group and Desktop Specifications is associated at the item revision level. Item VI1004 is assigned to the master organization Vision Operations and to Seattle Manufacturing.

2. Define the item pages for the attribute groups. You can specify one or more attribute groups on a page or create a separate page for each attribute group. Item pages are also inherited from the parent item catalog categories.

Associate item catalog attribute and create item page

In the figure below you can see that the Desktop catalog category has five item pages, one of which is the Benchmark Rating page containing the Benchmark Ratings attribute group.

Item pages for attributes on Item Detail page

Defining Item Attachment Categories

Whenever users add attachments to an item, they specify an *attachment category*. Item attachment categories provide a way to classify attachments—this classification identifies the purpose of the attachment to the business object, and can also be used as one of the search criteria within an attachment list. If users do not specify an attachment category for an item catalog category, and none are inherited from its parent catalog category, they always have the option of using the attachment category Miscellaneous.

> **Note:** Attachment categories inherited from a parent catalog category cannot be deleted at the child catalog category level.

Attachment categories are defined by the system administrator and are available across the various applications of the E-Business Suite. Attachment categories that you define can be made available to any E-Business Suite application. Thus, attachment categories are shared across the E-Business Suite applications and this should be kept in mind when defining any new attachment categories. Rules defining the usage of attachment categories are defined independently by each application.

You can define attachment categories in the Document Categories form. Before defining a new attachment category, verify that it does not already exist. If it already exists, then you can enable it for item catalog categories.

Defining an attachment category (for example, document category)
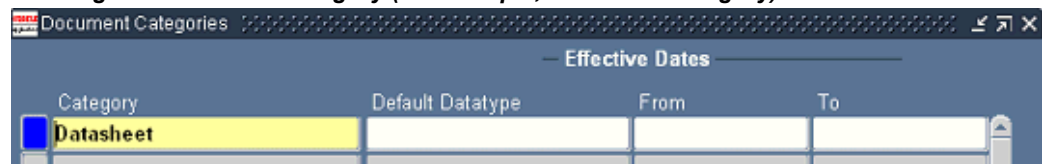
Associating Item Attachment Categories

Once you define an attachment category (also known as a "document category"), you can enable it for an item by associating it with an item catalog category.

All item catalog categories inherit attachment categories (in the same way they inherit item attributes) from their parent catalog category. So, before associating attachment categories with item catalog categories create a list of all attachment categories to be used in your item catalog. Consider which attachment categories are common among item catalog categories within each branch of the item catalog hierarchy. If an attachment

category is applicable to all the child catalog categories, then the attachment category should be associated with the parent catalog category.

When associating Attachment categories with item catalog categories, you can specify the effectivity of this association by providing a Start Date and an End Date. If an attachment category association has an end date, that type of attachment cannot be used for the items belonging to the category beyond the specified date.

See Also: For details about how to associate item attachment categories, see "Adding Attachment Categories to an Item Catalog Category" in the APC online Help (iHelp).

Associating an attachment category with an item catalog category

Setting Up Item Templates

You can define Item Templates for the Item Catalog to include operational attributes. You can also include user-defined attributes in an Item Template once it is associated with an Item Catalog Category. Item Templates may be inherited for use down through the item catalog hierarchy. You can specify a default Item Template for an Item Catalog Category. A reset is available to restore the default attributes and values inherited down through the item catalog hierarchy.

Benefits:

•   Default operational and user-defined attributes to speed up the process of specifying item attribute values

•   Enforce consistency of item attribute values for items belonging to different item catalog categories or item types. For example, you may have an item template for all purchased computer hard-drive assemblies.

How to Set Up Item Templates:

1.   Create Item Template in Setup Workbench to specify operational attributes

2.   Associate item templates with any item catalog category. Item templates are inherited down through the item catalog hierarchy

3.   Specify which item template is the default for the item catalog category

4. Specify values for any associated user-defined attribute groups to the item catalog category. When you define a template at a parent category, the same template is inherited by all child categories. You can modify attribute values in the inherited template and if needed revert to inherited values at a later time.

Updating item template for user-defined item category attributes

Items: Item Catalog Categories > Basic Information > Templates >

Update Template

Item Catalog Category:

Template **Purchased Item**
Description **Purchased Item**
Organization
☑ Set as Default

Attribute Group Operating Conditions ▾ (Go)
Multi Row **No**

Attribute Values

Select Attributes and ... (Enable) (Clear)

Select All | Select None

Select	Attribute	Value		Item Catalog Category	Enabled
☐	Optimal Temperature	100	Celsius ▾	Motherboards	☑
☐	Humidity	10	% ▾	Motherboards	☑
☐	Maximum Wet Bulb Temperature	20	Celsius ▾	Motherboards	☑
☐	Minimum Dew Point	65	Celsius ▾	Motherboards	☑
☐	Chassis Fan Speed			Motherboards	☐

Creating/updating item template for item master operational attributes

Template Details

Template Details

(Update)

Template **Purchased Item**
Description **Purchased Item**
Organization
Attribute Group Lead Times ▾ (Go)
Multi Row **No**

Attribute Values

Attribute	Value	Controlled At	Enabled
Preprocessing		Organization Item	No
Processing		Organization Item	No
Postprocessing		Organization Item	No
Fixed		Organization Item	No
Variable		Organization Item	No
Cumulative Manufacturing		Organization Item	No
Cumulative Total		Organization Item	No
Lead Time Lot Size		Organization Item	No

How to Use Item Templates

1. During the new item creation process you can specify an item template or use the default if one was setup at the item catalog category level

2. The operational attribute values specified in an item template are not set for 'unapproved' items (i.e. items requiring a new item request). The item template will need to be applied manually once the item is approved.

Defining Lifecycle Phases

An item lifecycle is a sequence of project phases. Each phase represents a set of tasks and deliverables that are required before promoting the item to the next phase. For example, the lifecycle phases for a computer component lifecycle might be: Concept, Design, Prototype, Pre-Production, Production, and Retirement. If you are using Oracle Projects you can associate the item lifecycle with a project based on a project template or an existing project. The project workplan's top tasks can be mapped to the lifecycle phases.

> **Important:** If you are using a lifecycle tracking project, then set the current phase within the project workplan. This action will automatically update the item phase.

Define lifecycle phases in Oracle Projects. For more details about defining lifecycle phases, see the *Oracle Projects User's Guide*.

Defining lifecycle phases

Specify the valid item statuses for each lifecycle phase. You can select a default item status for each phase. When you promote an item to the next phase the default item status is automatically applied to the item. These item statuses set operational characteristics of the item (such as whether the item is Purchasable, Transactable, Stockable, or whether BOMs are allowed). For example, in the Concept and Design phase you may have a default status of Inactive, so you cannot transact the engineering item in inventory or purchasing. However, in the Pre-Production and Production phases you may specify that the default item status is Active so that it is fully transactable in the E-Business Suite.

See Also: For more details about associating lifecycles to item catalog categories, see the following topics in the APC online Help (iHelp):

• Associating Lifecycles with an Item Catalog Category

- Managing Lifecycle Change Policies
- Managing Lifecycle Phases

Valid item statuses for lifecycle phase

Phase Details

Display Name **Pre-Production**   Description **Production trial runs, validation testing, release to manufacturing**

Effective Start Date **01-Jun-2003**   Effective End Date

Valid Item Statuses

Item Status	Description	Default
Active	All functions enabled	✔
Engineer	Engineering item	🖼

Item Statuses

Item Status Details

Status Information

                                                                    [Update]

Status **Active**   Description **All functions enabled**
Inactive Date

Status Attributes

Attribute	Value	Usage
BOM Allowed	Yes	Default control
Build in WIP	Yes	Default control
Customer Orders Enabled	Yes	Default control
Internal Orders Enabled	Yes	Default control
Invoice Enabled	Yes	Default control
Purchasable	Yes	Default control
Stockable	Yes	Default control
Transactable	Yes	Default control

Additional Attributes

Defining Lifecycles

You can create lifecycles to manage the lifecycle phases of your items and item revisions. For example, you may have one lifecycle at the item level to manage new part introductions while having another lifecycle to manage the part revision process. You can use different lifecycles to manage manufactured assemblies and outsourced components.

Once the lifecycle phases have been defined, you create a lifecycle in Oracle Projects. To associate a lifecycle with an item or item revision you must check Enable for Items on the **Lifecycle Details** page. You must also check Enable for Projects if you use Oracle Projects.

If you intend to associate lifecycle tracking projects with your item lifecycles, then when setting up your project or project template do the following.

- Assign the lifecycle to a work plan for the project or project template
- Assign a lifecycle phase to each top task.

See Also: For more details about defining lifecycles, see the Oracle Projects Users Guide.

Defining Lifecycles

Lifecycle Details

* Indicates required field
  * Name **Computer Component Lifecycle**
* Short Name **Computer Component Lifecycle**
  Description **Computer component item lifecycle**

☑ Enable for Projects
☑ Enable for Items

Lifecycle Phases

Ⓢ Previous Next 5 ⊗

Sequence Number	Phase	Short Name	Name	Description	Delete
1	Concept	Concept	Concept	Evaluate market, define requirements	🗑
2	Design	Design	Design	Define specifications, structures, and processes	🗑
3	Prototype	Prototype	Prototype	Build and test prototypes	🗑
4	Pre-Production	Pre-Production	Pre-Production	Production trial runs, validation testing, release to manufacturing	🗑
5	Production	Production	Production	Begins immediately with the production cutover, marking the last phase of the implementation, and the beginning of the system support cycle	🗑

Associating Lifecycles to Item Catalog Categories

Associating a lifecycle with an item catalog category makes the lifecycle available for all items in the item catalog category. Lifecycles are inherited throughout the item catalog category hierarchy—child categories inherit the lifecycle from the parent category. The following figure illustrates the valid lifecycles for the Motherboard item.

Associating lifecycles with item catalog categories

Lifecycles

Item Catalog Category: **Motherboards**

(Add)

Select Lifecycle Associations and... (Delete)

Select	Name	Description	Classification
⊙	Computer Component Lifecycle	Computer component item lifecycle	Motherboards
○	Computer Component Revision Lifecycle	Computer component item revision lifecycle	Motherboards

(Add)

Change Policy

▶ Show Key Notation

View Policy for Attributes ▼ (Go)   Update Concept ▼ (Go)

Attribute	Policy for Associations / Policy for Attachments / Policy for Attributes / Policy for Promote and Demote	Concept	Design	Prototype	Pre-Production	Production	Retirement
Memo...		✓	✓	📑	📑	📑	✗
Memo...		✓	✓	📑	📑	📑	✗
Assembly Clearances		✓	✓	✓	✓	✗	✗
Benchmark Ratings		✓	✓	✓	✓	📑	✗
Operating Conditions		✓	✓	📑	📑	✗	✗
Motherboard Specifications		✓	✓	📑	📑	📑	✗

Change Policy

For each lifecycle associated with an item catalog category, a Change Policy can be defined as detailed under Defining Change Policy. You can specify for each lifecycle phase of the item whether a change is allowed or not allowed or is possible only through a change order.

Defining Change Policies

Administrators can define and update item change policies for item catalog categories. These item change policies determine the rules for how and when an item's attributes, attachments, and associations are changed. For example, a company manufacturing an engine that has hundreds of specifications may wish to define item change policies for the different phases of development. So when the engine is in the concept or design phase, many of the attributes are allowed to change without formal approval, and the lifecycle policy is not very restrictive. When the engine progresses to the prototype phase, the company might wish to place all attributes, attachments, and associations under stricter change control, perhaps requiring a change order for all modifications. Later, when the engine is in the production phase, the company will likely want every facet of the engine's production under tight control.

Defining item change policies enables a company to:

- Specify whether or not item changes are allowed in a particular lifecycle phase of an item lifecycle

- Specify what types of changes (attributes, attachments, associations) are allowed for an item in each lifecycle phase

- Specify whether or not a change order is required to make particular changes to an item in a specific lifecycle phase

    > **Note:** Change policies inherited from a parent catalog category cannot be edited at the child catalog category level. If you do not define a change policy for an item, then by default changes are allowed in all lifecycle phases. Item change policies only take effect after the item has been approved.

On the Update Change Policy for Attributes (Associations or Attachments) page, select the appropriate attribute group (association or attachment category) and the change policy you wish to apply for the particular phase. The valid change policies are:

- Allowed

- Not Allowed

- Change Order Required

Setting lifecycle change policy

> **Note:** Before establishing a change policy for attributes or attachments, you must first associate the attribute group or attachment category with the item catalog category.

Defining New Item Request Types

New Item request is a workflow process, which allows you to route the definition and approval of a new item in a step-by-step wizard process. When creating a new item, various aspects of an item like base attributes, user-defined attributes, AML etc. could be defined by various people in the organization using a workflow process. The definition routing and approval routing could be combined in a new item request workflow.

New Item Request (NIR) is a Change category; there could be many types of New Item Request that suits a particular business process. Each type of NIR allows you to specify NIR number generation function, default assignee type and assignee. You can create a NIR type for a outsourced components which will have a specific numbering prefix and assigned to person with a role at the item category level.

Defining New Item Request Type

Once a specific NIR type is created, the administrator can assign additional information like the NIR related attributes to be associated. Examples of such attributes are Cost Impact, Implementation costs etc. These attribute groups then could be organized in different pages similar to the Associating Item Catalog Attributes.

Defining User defined Attributes for New Item Request Type

Defining New Item Request Workflows

One of the most important functionality in NIR type, is to be able to associate workflow. The administrator can add workflow steps and specify the status of the NIR in which the workflow will be executed and assign the workflow, by choosing it from a list of workflow template. The administrator should specify the step number, status of the NIR process in which the workflow will be initiated and finally associate the workflow by choosing it from the workflow templates.

Defining a workflow template that could be assigned to a New Item Request is detailed in Defining Workflow Templates section later in the document. NIR workflow are typically Definition and Approval Workflow, an example of NIR workflow is outlined in the following figure.

NIR Workflow example

Once you associate the workflow template with the NIR type, you need to specify the category association for the NIR.

Setting Up New Item Request for an Item Catalog Category

To associate a specific New Item request type with a category, you need to navigate to category details and specify the New Item request type in the New Item Request page.

In the Workflow/Approval section, you can specify which people are required to enter attributes during the workflow process for each lifecycle phase of the new item request. For example, when the new item request is in the Open phase, one person in the workflow may be responsible for some technical specifications associated with the new item being created; another might need to provide details about the new item's safety standards--in this case you would be sure to associate the attribute groups related to technical specifications and safety standards so that the appropriate people can add the information necessary to define all aspects of the new item.

Since NIR is change, it has a status type. Only the certain workflow type is allowed for a given NIR status type as tabulated below.

NIR Status Type	Valid Workflow Type
Open	Definition
Approval/Review	Definition and Approval, or Approval
Others	Generic

Associating a New Item Request with an Item Catalog Category

Associating Item Attribute groups with Definition Steps

Associate Attribute Groups

• Indicates required field   (Cancel) (Apply)

Step **20**
Workflow Process **Definition and Approval**
Assignee Type **Item Role**
Name **CAD Librarian**

•Attribute Groups		Delete
Technology	✎	🗑
Layout Specifications	✎	🗑
Symbol	✎	🗑
Footprint	✎	🗑
(Add 5 More)		

(Cancel) (Apply)

Defining Item Number and Description Generation Rules

For a given category, item number and description could be automatically generated when an user wants to create an item. Various item number and description generation methods are supported such as:

Sequence Generated

If the item request numbering method is sequence generated, then all item number requests will be automatically generated based on the sequence generation rules specified here. You can specify a prefix, starting number, increment, and suffix.

> **Note:** Sequence generation does not apply to Item Description generation--it is for item number generation only.

Function Generated

If the item request numbering/description method is function generated, then all number/descriptions for new items in the item catalog category will be automatically generated based on a custom function. When specifying that an item number/description is function generated, select the function that you want to use, and then map the function parameters to the user-defined attributes that you want included in the item number/description. When you create a new item, users will be prompted to enter the user-defined attributes used in generating the item number/description.

> **Note:** You must have already defined a custom function before using this item number/description generation method.

Inherited from Parent

If the item request numbering/description generation method is inherited from parent, then the item number/descriptions for the items will be generated using the same method as the parent item catalog category.

> **Note:** All item catalog category subcategories have, by default, a number/description scheme that is inherited from parent, which could be overridden at the child level.

User entered

If the item request numbering/description method is user entered, then the user is required to manually enter a number or a description for new items.

Item Number Generation

Auto Generation

Catalog Category: **Capacitors**

Number Generation

Item Numbering **Sequence Generated**

Prefix VI

Starting Number 60000

Increment 1

Suffix CAP

(Update)

Setting up Item Number Generation

Item Number Generation

(Cancel) (Apply)

Catalog Category : **Capacitors**

Item Numbering Sequence Generated ▾ (Go)

Function Generated
Inherited From Parent
Sequence Generated
User Defined

Prefi

* Starting Number

* Increment 1

Suffix CAP

(Cancel) (Apply)

Description Generation Sample

Auto Generation

Catalog Category: **Capacitors**

Description Generation

Item Description **Function Generated**

(Update)

Generation Function

* Indicates required field

Function **GenCapacitorItemDesc**

Mappings For Function

Function Parameter	Attribute Group	Attribute
Tolerance	Capacitor Specifications	Tolerance
Capacitance Value	Capacitor Specifications	Capacitance Value
Size	Capacitor Packaging Specs	Size
Package Type	Capacitor Packaging Specs	Package Type
Case Size	Capacitor Packaging Specs	Case Size
Dielectric	Capacitor Specifications	Dielectric
Voltage	Capacitor Specifications	Voltage

Defining Item Catalog Search Criteria

Criteria templates offer you a way to save frequently used search criteria--essentially, acting as a saved list of specific attributes and attribute values. Criteria templates that you—the administrator—create are available to all users, and serve to expedite searches for an item, thereby saving users the time it would take to specify search criteria and execute frequent item searches.

Of course, users also have the option of creating criteria templates for their own private use. In fact, when both a user and administrator have defined a default criteria template for a particular item catalog category, the user-defined criteria template takes precedence.

Because criteria templates contain user-defined attributes, you must always define them in the context of an item catalog category. You can define as many criteria templates

as you wish, and you can also denote the most commonly used as the default criteria template.

The following figure shows the **Advanced Search** page, where you would define a criteria template for the item catalog category Capacitors. Begin by listing the common attributes a typical user might search for when using the item catalog category Capacitors. Here, attributes like Capacitor Voltage and Capacitor Value, among others, are specified. After adding attributes, you can specify operators and values to further restrict the scope of your search.

See Also: For more details about creating item catalog criteria templates, see the APC online Help (iHelp).

Defining item catalog Search Criteria

Defining Item Catalog Display Formats

Results Formats enable you to predefine search results views for each item catalog category. You can use these views to look at different sets of attributes of the items that are returned by a search. Both administrators and users can create results formats. Administrator-created results formats are available to all users. User-created result formats are available only to the users who created them. You can define as many results formats as you wish, and you can also denote the most commonly used as the default results format.

You cannot use results formats to display multi-row attribute groups and attributes set at the item revision level.

You can define a results format by listing the common information a typical user might wish to view for an item. For example, for the item catalog category Capacitor, you might select Item Name, Description, Item Status, or other important item attributes such as Capacitance Value, or Tolerance.

See Also: For more details about creating item catalog results formats see the APC online Help (iHelp).

Defining Item Catalog Import Formats

An import format identifies the base and user-defined attributes (but not the operational attributes) in an item catalog category that is imported into APC via a spreadsheet. You cannot import operational attributes; however, you can specify an item template and item status in your import format. Both item templates and item statuses can be used to set operational attributes when importing items.

The base item attributes are:

- Item Name
- Item Catalog Category
- Description
- Primary Unit of Measure
- Lifecycle
- Lifecycle Phase
- Item Status
- Long Description

You can define import formats for both items and item revisions. An import format can also import values for a multi-row attribute group.

Defining Report Templates Using XML Publisher

Creating a template file consists of two basic steps:

1. Design your template layout - Use the formatting features of a word processing application and save the file as RTF.

2. Mark up your template layout - Insert the XML Publisher simplified tags.

To create a template, you need to create or use any of the existing seeded data definitions and associate or assign the change template file with the corresponding data definition file. For example, if you have a Change Order template file, associate it to a Change Order data definition file.

When you create a template, you need to assign it a data definition and upload the RTF. There are seven seeded "Summary" templates available in the system. These cannot be updated. Initially, upload one template file for a specific language and territory combination. This file will become the Default Template File. To upload additional template files or to change the Default Template File, use the **View Template** page.

To create a template:

1. Using the XML Publisher Administrator responsibility, navigate to the **Templates** page.

2. Select the Create Template button

3. Provide the following information:

   Name

   Enter a user-friendly name for your template.

   Code

Assign a template code using the product short name and a descriptive ending.

Application

While creating templates, select the correct application from the LOV. This application should be the same as the application of the Data definition. For Change Templates, the supported application is "Engineering".

Data Definition

Select the product's data definition from the LOV. Oracle PLM comes seeded with seven data definitions (Issue, Idea, Change Request, Change Order, New Item Request, Attachment Approval, Attachment Review). For example for an Issue Report Template, select the seeded Data Definition for Issues. This will result in the Template being available for Issue Reports. Type Select the input template type (RTF, PDF, or XSL-FO) from the LOV.

Start Date

Enter the date from which the template will be active.

End Date

To make the template inactive, enter an end date.

Subtemplate

If this is a subtemplate, select the check box.

> **Note:** A subtemplate is referenced by other templates, but cannot run on its own

File

Use the Browse button to upload your RTF or PDF template layout file.

Language

Select the template language from the LOV. Add more language template files to your template definition from the View Template page.

Territory

Select the language territory from the LOV.

> **Important:** After the template definition is created, the following fields cannot be updated: Application, Code, and Type. You can update the template from the View Template page.

Example of RTF Template file for Motherboard Items
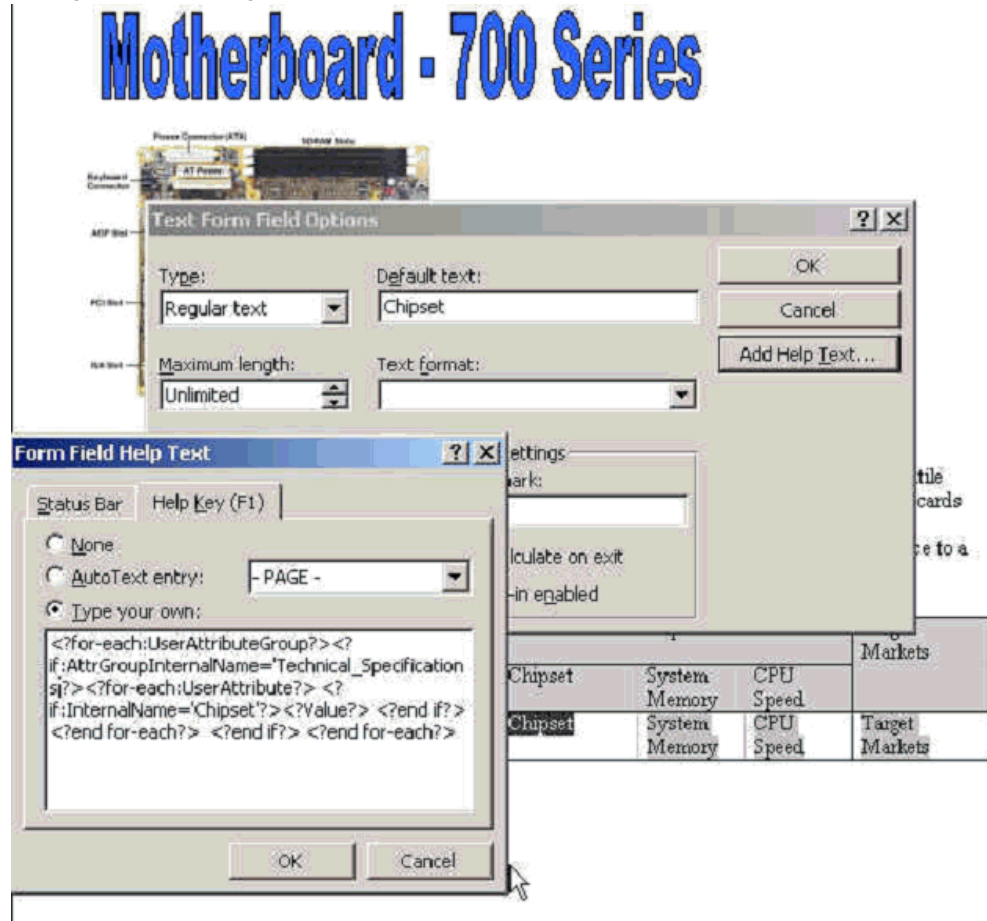
In the template file, you can include simplified XML tags to refer the Item information you want XML Publisher to print while generating the report.

Here is an example of the XML Input file that will generate report for Motherboard Items using the RTF template.

Item Primary Information

Item Attribute Information

For further information, see:

- *Oracle XML Publisher User's Guide*
- *XML Publisher Feature Listing*

Associating Report Templates to Item Catalog Categories:

Associating a Report Template with an item catalog category makes these templates available for all items in the item catalog category. Report Templates are inherited throughout the item catalog category hierarchy—child categories inherit the Report Templates from the parent category. The following figure illustrates the valid Report Templates for the Motherboard item.

Associating Report Templates with an Item Catalog Category

> **Note:** For details about how to associate Report Templates, see "Associating Report Templates to an Item Catalog Category" in *Oracle Product Lifecycle Management User's Guide*.

Defining Cross Reference Types

Item can have many cross-references. Example of cross-references are, customer part number, superseding/preceding item number etc. An administrator can define their own cross-reference types based on their business need. User-defined flex-fields and value sets can be associated with each cross-reference types. Cross-reference types have effectivity dates, which controls them being active or inactive.

Defining Cross Reference Types

4

Defining Catalogs

Overview of Catalogs

Defining catalogs includes the following tasks.

Tasks	Required?
Defining Catalog Categories	Yes
Defining Catalogs	Yes
Implementing Catalog Security	Yes

A *catalog* is a collection of items classified within a hierarchical taxonomy. A catalog consists of a hierarchy of catalog categories, each containing items assigned to the catalog category. Catalogs provide a convenient way to organize and present your items under various taxonomies. For example, you can set up a Product Catalog of sales items that the Sales and Marketing organizations, as well as your customers, can browse to find products. You can set up a Purchasing Catalog, which contains all the purchasable items that the Engineering, Manufacturing and Procurement organizations can browse to locate parts. You can also set up a Service Catalog that contains all serviceable items and service products for your field repair and customer service departments. The following figure shows the catalog hierarchies for a Product catalog, Purchasing catalog and Service catalog. You can define any number of catalogs and assign an item to more than one catalog. Depending on your setup, you can also assign an item to more than one catalog category within a catalog.

> **Note:** The *category sets* defined in the E-Business Suite appear as *catalogs* in APC. The categories within each of these category sets appear as catalog categories in the respective catalog.

Example of catalog hierarchies

Product Catalog	Purchasing Catalog	Service Catalog
▼ Computer Systems	▼ Electromechanical Components	▼ Computer Parts and Components
Desktop Products	Audio Devices	Hard Drives
Laptop Products	Circuit Breakers	Monitors
▼ Computer Parts and Components	Fans	Holders
Motherboards	Industrial Controls & Sensors	Mounts
PCMCIA	Knobs & Dials	Memory
Keyboards	▷ Power	Motherboards
Mouse	Relays & I/O Modules	PCMCIA
Memory	Switches	Keyboards
Hard Drives	▼ Electronic Components	Mouse
Monitors	▼ Discretes	▷ Electromechanical Components
Holders	Diodes	▷ Mechanical Components
Mounts	Gates	▼ Networking Components
	Rectifiers	Ethernet Controllers
	Transistors	Ethernet Cards
	▼ IC's	▼ Consulting Services
	Memory	Enterprise Solutions
	Analog IC's	Small Business Solutions
	Digital IC's	Helpdesk Outsourcing
	Microcontrollers	▼ Support
	Microprocessors	▼ Business Support
	▷ Optoelectronics	Gold Support
	▼ Passives	Silver Support
	Capacitors	Bronze Support
	Circuit Protection	▼ Consumer Support
	Encoders	On Call Support
	Frequency Controlers	Warranty
	Inductors	Extended Warranty
	RC Networks	
	Resistors	
	Signal Transformers	
	▷ Interconnect components	
	▼ Mechanical Components	
	Enclosures	
	Fasteners	
	Heat Sinks	
	▼ Networking Components	
	Ethernet Controllers	
	Ethernet Cards	
	▷ Test, Tools & Supplies	

Defining Catalog Categories

You use a *catalog category* to classify items within a catalog. You can define unlimited catalog categories and reuse catalog categories in more than one catalog. Before creating catalog categories, consider all the catalogs you need to define throughout your enterprise (for example, Product catalog, Purchasing catalog, Service catalog). Then create a list of all the catalog categories and sub-categories you will need to correctly classify each item within your catalogs. Can you reuse the same catalog category in

several catalogs? For example, you may have a catalog category for Hard Drives in each of your catalogs since you sell hard drives to customers, purchase hard drives from your supplier, and service hard drives. You should use the same flex structure for catalog categories so you can reuse them in different catalogs.

See Also: For more details about defining catalog categories, see "Creating Categories" in the APC online Help (iHelp).

Defining Catalogs

A *catalog* is a hierarchy of catalog categories used to classify items defined in your item catalog. Once you have defined all of your catalog categories you can create each of your catalogs. When you construct your catalog category hierarchy keep in mind that only the leaf nodes of the catalog can contain items.

You can set up catalogs in a way that the items may be classified the same across all organizations or differently between organizations. You can enable item assignments to more than one catalog category within your catalog.

See Also: For details about defining catalogs, see the following topics in the APC online Help (iHelp):

- Creating Catalogs

- Assigning a Category to a Catalog

- Creating a Catalog Category within a Catalog

Implementing Catalog Security

You can use role-based security to control which users, groups, companies, or organizations can view and manage catalog items. For example, you may allow customers and the Sales organization to view your Product catalog, while only Engineering, Procurement, and Manufacturing can view the Purchasing catalog.

There are two privileges that you can grant a user with a role on the catalog: View Catalog and Manage Catalog. The two seeded roles are Catalog Viewer and Catalog Manager. Users who have been granted the View Catalog privilege can browse and view all items in the catalog. A user granted the Manage Catalog privilege can assign items to a catalog category within the catalog.

Role	View Catalog Privilege	Manage Catalog Privilege
Catalog Viewer	Yes	No
Catalog Manager	Yes	Yes

See Also: For details about implementing catalog security, see the following topics in the APC online help (iHelp):

- Assigning People to a Catalog

- Assigning People to a Valid Category

5

Defining Change Categories and Types

Overview of Change Categories and Types

You must perform the following tasks to define Change categories and their associated types.

Task	Required?
Defining Change Categories	
Defining Change Types	Yes
Defining Line Types	
Defining Change Type Attribute Groups	
Defining Change Line Attribute Groups	
Setting Up Change Attribute Group Security	
Implementing User-Defined Functions for Change Type Attributes	
Associating Change Type Attributes	
Customizing Change Management Workflows	
Defining Approval Routing Templates	Yes
Creating User-Defined Priority Codes	Yes
Creating User-Defined Reason Codes	
Creating User-Defined Statuses	
Defining Change Category Criteria Templates	
Defining Change Category Results Formats	
Defining Change Reports	

Defining Change Categories

You can use *change categories* to define and manage changes required by your enterprise. In addition to the seeded change categories (issues, change requests, change orders) you can create change categories specific to your business needs. For example, you can create the change category Enhancement Requests to track your

customers' product enhancement requests. You cannot delete seeded change categories; however, you can disable them.

You can configure each change category to have revised items or request lines based on the business purpose of the category. For example, a Change Order can have revised items as they allow a change order to implement item related changes. Request lines enable you to request changes or specify tasks related to an item and assign it to a person or group. You can associate criteria templates and result formats to a change category for frequently executed search criteria.

APC seeds the following change categories:

Issues To track, manage and resolve various product/process related issues

Change Requests To request changes and get approval for requested changes

Change Orders To implement requested changes and revise items

IdeasTo capture suggestions, innovations, improvements etc. from customers and internal users

New Item Request To provide a formal process of definition and approval of a new item

Document Review To allow for an informal process of review and feedback on documents

Document Attachment Approval To allow for a formal process of review and approval of documents

You can specify a form function to control who can view and create change category objects. The form function must be added to the change management security menu (EGO_CHGMGMT_USER_SECURITY), which in turn is referenced by a user responsibility (such as Development Manager or Development Engineer). You can also enable (in the Item Catalog workbench) a tab to show all instances of a change category for the item in context.

New change category Enhancement Requests enabled in the Item Catalog

Enabling Change Category Function Security and the Item Change Category Tab

To enable the change category function and the item change category tab:

1. Create the Form Function for the Item Enhancement Request tab.

Description Tab

Function

EGO_ITEM_ENH_REQ

User Function Name

EGO User Item Enh Request tab

Description

EGO User Item Enh Request tab

Properties Tab

Type

SSWA JSP FUNCTION

Maintenance Mode Support

None

Context Dependence

Responsibility

Web HTML Tab

HTML Call OA.jsp?page=/oracle/apps/ego/item/eu/webui/
EGOTEMCHANGEMGMTLISTPGL=431=RP=EGO_USER_WORKBENCH_HOME
PAGE=**EGO_ITEM_ENH_REQ=ENHANCEMENT_REQUEST**

> **Important:** Boldfaced characters represent the form function and
> internal name of the change category, respectively.

2. Create the Form Function for Enhancement Request security. The function name must
   be 'ENG_CREATE_' (for example, ENG_CREATE_ENHANCEMENT_REQUEST):

Description Tab

Function

ENG_CREATE_ENHANCEMENT_REQUEST

User Function Name

Engineering Create Enhancement Request

Properties Tab

Type

SSWA JSP FUNCTION

Maintenance Mode Support

None

Context Dependence

Responsibility

Web HTML Tab

HTML Call OA.jsp?page=/oracle/apps/eng/changemgmt/webui/SelectChgTypePG=
ENHANCEMENT_REQUEST

> **Note:** Boldfaced characters represent the internal name of the
> change category.

***Create form functions for change category security and item change
category tab***
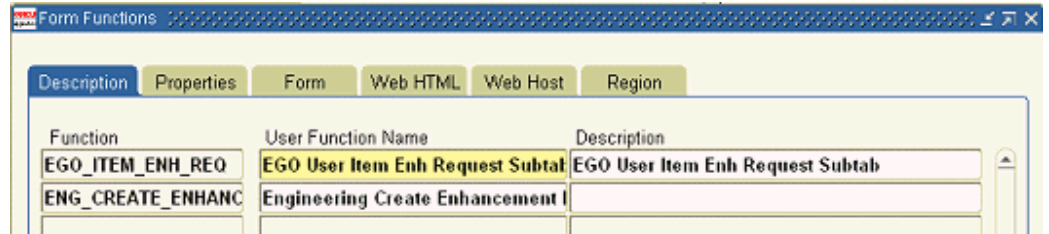
3. Add the Create Enhancement Request function to the change management
   navigation menu (see the menu EGO_CHANGE_MGMT_MENU).

   Prompt

   Create Enhancement Request

   Function

   Engineering Create Enhancement Request

   Add function to change management menu
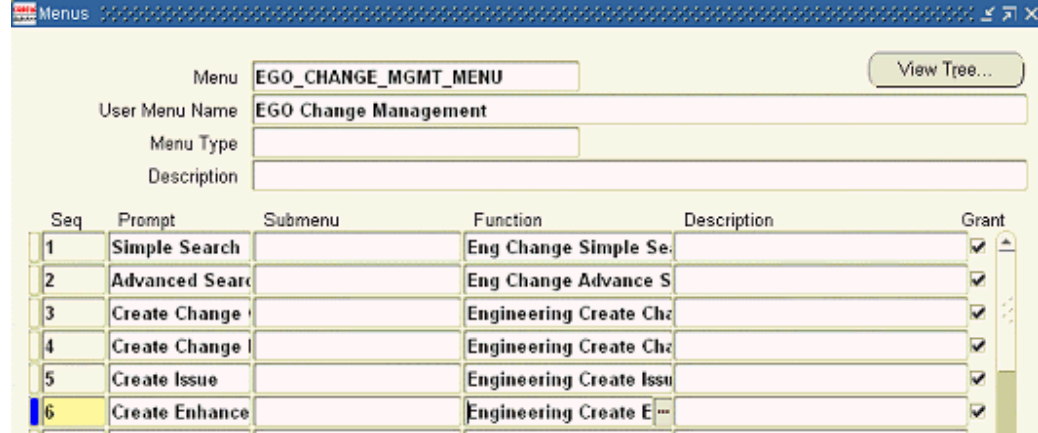
4. Add the Enhancement Request security function to the change management security
   menu (see the menu EGO_CHGMGMT_USER_SECURITY).

   Function

   Engineering Create Enhancement Request

Add function to change management security menu
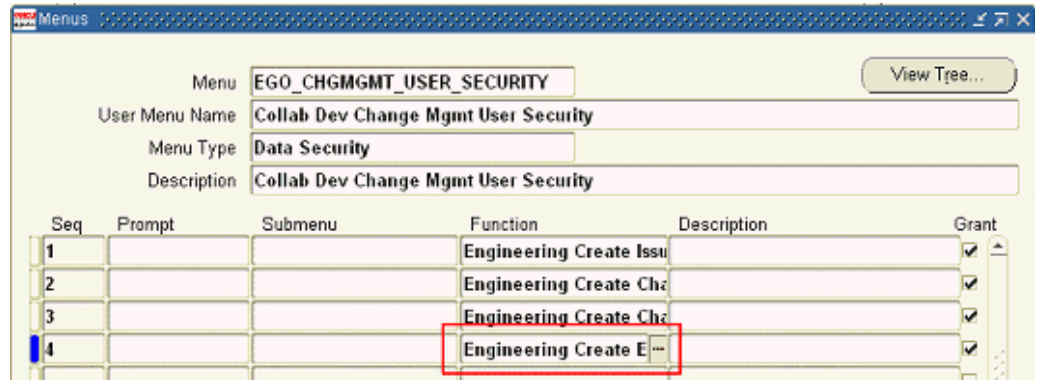
5. Add the Item Enhancement Request function to the item change management tab menu (query the menu EGO_USER_CHANGE_MANAGEMENT_TAB)

Prompt

Enhancement Requests

Function

EGO User Item Enh Request tab

> **Important:** You must enter a prompt when adding a function to a change management menu such as EGO_USER_CHANGE_MANAGEMENT_TAB.

6. Add the new change category for Enhancement Requests by duplicating an existing Change Category. In example shown in the new Enhancement Request Change Category is being defined by duplicating the Issue category:

Internal Name

ENHANCEMENT_REQUEST

Name

Enhancement Request

Description

Enhancement Request

Plural Name

Enhancement Requests

Sort Seq

9

Start Date

defaults to sysdate

Number Generation

Sequence Generated

Prefix

ER-

Next Available Number

0001

Add new change category by duplicating an existing change category

Duplicate Category: Issue
* Indicates required field   (Cancel) (Apply)

* Internal Name   ENHANCEMENT_REQUEST
* Name   Enhancement Request
Description   Enhancement Request

Plural Name   Enhancement Requests
Base Change Category **Issue**
* Sort Sequence   9
* Start Date   01-Apr-2005
                   (example: 17-Mar-2005)
End Date
                   (example: 17-Mar-2005)

Autonumbering
* Number Generation   Sequence Generated
* Prefix   ER-
* Next Available Number   0001

                                                                (Cancel) (Apply)

> **Note:** Oracle recommends that you stop and then start the Jserv and
> Apache listener middle tier ports after completing these setup tasks.

For more information on change category administration, see "Creating Change
Categories" in the APC iHelp. For more information about form functions, see the Oracle
Applications System Administrator's Guide and Oracle Applications Developer's Guide.

Creating Priority Codes

Priority codes enable you to track the urgency of an issue/change. You can create priority
codes to capture different priorities (such as High, Medium or Low).

Priority codes are applicable to all change categories and their types.

You cannot delete seeded priority codes; however, you can disable the seeded priority
codes and define new ones specific to your use. You can disable a priority on a given
date by specifying a date in the Inactive On field.

Creating Reason Codes

Reason codes enable you to track the reason for which the issue/change has been
created. Create reason codes to capture reasons for the issue/change (such as
Quality Improvement, Design Improvement, Cost Reduction, Test Failure and Non
Conformance).

Reason codes are applicable to all change categories and their types.

You cannot delete seeded reason codes; however, you can disable the seeded reason codes and define new ones specific to your use. You can disable a reason on a given date by specifying a date in the Inactive On field.

Creating Classification Codes

Classifications provide a mechanism for companies to automate the categorization of change orders, and to also indicate to users exactly how the change order will impact their production. APC provides two types of classifications:

Derived

Derived classification codes are derived from a user-defined function. For example, a division of a company, Vision Operations, needs to automate the process whereby a change order is assigned a particular classification code. To create an automated classification process, Vision has created a set of attributes that, when filled in by users, classifies change orders into a particular classification or workflow routing. The user-supplied attributes are mapped to user-defined functions. The function takes the data supplied in the attributes and derives a valid classification code. Derived classification codes appear to users as read-only data. For details about user-defined attributes and functions see Defining Header/Line Type Attributes and Attribute Groups, and Defining User-Defined Functions. For details about setting up user-defined functions for change type attributes, see the example: Implement User-Defined Functions for Change Type Attributes.

Valid

Valid classification codes are selected by the user from a list of values. Valid values are specified in the change header type. Note: Classifications are available only to change categories whose base category is Change Order.

Creating Statuses

Statuses enable you to manage an issue/change through its lifecycle. You define statuses to indicate various states of an issue/change (for example, Open, On Hold, Complete and Cancelled).

Statuses are applicable to all change categories and their types

You cannot delete or disable seeded statuses; however, you can define new statuses specific to your business processes. You can disable user-defined statuses on a given date by specifying a date in the Inactive On field.

Defining Workflow Templates

Workflow templates enable you to predefine a business approval process. Change categories such as issues, change requests and change orders can only be approved via the successful completion of an approval routing. You can create and maintain workflow templates for each change type using workflow templates listed in the Setup Workbench under the Change Management Workflow tab.

Creation of a new Workflow Template

While creating a workflow template its Type needs to be specified.

> **Important:** Once a workflow template is created of a particular type, its type cannot be changed.

Currently the following workflow template types are supported:

Approval

The Approval workflow template type is valid only for workflows with status type Approval.

Definition

The Definition workflow template type is used primarily for workflows in New Item Requests with status type Open. Definition and Approval The Definition and Approval workflow template type is used primarily for New Item Requests of status type Approval.

Generic

The Generic workflow template type is used for all other status types.

Certain workflow types can be associated with specific statuses. Following are the associations between status type and workflow types:

Valid Workflow Types for Change Categories (except New Item Request)

Status Type	Valid Workflow Type
Approval/Review	Approval
Others	Generic

Status Type	Valid Workflow Type
Open	Definition
Approval/Review, Definition and Approval, or Approval	Approval
Others	Generic

Workflow templates are basically made up of approval steps; each approval step describes a workflow process and specifies the assignees. For example, you can create steps to request approval, request comment, or send an FYI notification.

You can use any of the following seeded workflow processes in a workflow template:

Request Approval

The Request Approval workflow enables you to request approvals from a person or group.

FYI

The FYI workflow enables you to send an FYI notification to a person or group.

Request Comment

The Request Comment workflow enables you to request comments from a person or group.

Definition

The Definition workflow is primarily used in the New Item Request Process and allows for the association of item attribute groups in the New Item Request process so that they can be defined by the step assignee.

Definition and Approval

The Definition and Approval workflow is similar to the Definition workflow but also requires an approval by the step assignee. For more details please refer to Defining New Item Request section.

You should plan and document the approval processes needed for specific types of changes in your enterprise. Pre-planning of these processes enable you to define workflow templates that adhere to your business processes.

You can manage parallel and serial approvals by assigning multiple people or groups to a routing step, or by assigning individual people or a group to each routing step. You can assign a specific role, person, or group to a routing step. Item roles need to be mapped to the change roles assigned on an approval routing step to ensure a person or group is assigned when the change is created (see " Defining Roles" for details on mapping item roles to change roles).

Customizing Change Management Workflows

Oracle Corporation supports the extension of workflows. Extensions include using existing seeded processes to build new workflow processes and modifying parameters of an activity without changing process logic (for example, adding custom logic in an Abort Approval Routing subprocess).

If the Change Management seeded workflows do not meet your business processing needs, you can customize or extend the seeded Change Management Workflow processes by creating your own processes, modifying the seeded ones, adding a new activity to a seeded process, and/or modifying custom hook PL/SQL procedures.

See Also: For details about customizing the seeded workflows or adding new workflows to Oracle APC, see "Appendix A: Change Management Workflows."

Suggestions for Customizing Workflows

- If you decide to create a new workflow process for Priority Change rather than modifying the seeded workflow, you must write your own custom logic to start your custom workflow in the custom hook called in Start Workflow.

- If you add a new workflow process in the Change Approval Routing Step (ENGCSTEP), then the process will be enabled as an Approval Routing Step Workflow in the Approval Routing pages. The workflow process is automatically called from the Approval Routing workflow.

- The activity will be shared by more than one object's process, so create the activity in the ENG: Standard (ENGWFSTD) item type. Otherwise, create the activity in the appropriate object's item type.

- Instead of modifying workflow notifications, you should create new messages. Oracle Workflow Builder does not maintain version information for objects such as item types, item type attributes, messages, and lookup types. For these objects, the latest definition always applies, so you must consider whether a change to any of these objects is backward compatible. If the modification affects existing processes, you should create a new object rather than edit the existing object.

    Important: Oracle Corporation provides support only for its seeded activities, processes, and the types of extensions described in this guide. Oracle Corporation does not provide support for your custom activities and processes.

Creating Task Templates

Task templates define the various tasks that must be completed for a change order. You can create task templates only for categories whose base change category is Change Order.

Creation of a Task Template

The task templates are created for specific Organizations. Hence multiple task templates could be associated with a change order type. Depending on the organization in which the change order is created, the task templates defined for that organization associated with the change order become applicable.

If a task is specified as Mandatory it means that the task must be completed before the change order can proceed to the next status. After creating task templates, you can use them to define the organization policies for change order header types. This has been detailed in the section Defining Header Types.

Defining Header/Line Type Attributes and Attribute Groups

Defining attributes and attributes groups enables you to capture additional information related to issues, change requests, and change orders. You can create user-defined attributes with validation logic and associate them to change types as a collection of attributes within an attribute group.

Attributes are defined by their names and values, and are saved within attribute groups. You can associate attribute groups to a change header type or change line type. You must define separate attribute groups for header types and line types. You can reuse the same attribute group across different change categories and their change types. Users enter the values for the attributes on the pages that you create for each change type. You can also define how attributes are displayed for your change objects to improve usability.

You can index attributes to speed up search performance. For numeric or date data types, a B-tree index enables users to search on a range of values or use relational operators such as "less than" and "greater than," among others. For text attributes, Oracle Text index allows flexible key word searching.

Prior to creating user-defined attributes, do the following:

- Group related attributes within the same attribute group. The following table shows some examples of attribute groupings:

Attribute Group	Attributes	Data Type
Customer Priorities	Customer	Char
	Priority	Char
	Date	Standard Date
	Review	Char
Implementation Cost	Manufacturing	Number
	Engineering	Number
	Supplier Charges etc	Number

- Create an object role (change) with custom privileges to define attribute group security (if necessary). You can secure the attribute group by setting Edit/View privileges. Later on, only users with certain roles can view or edit those attributes.

- Determine the data type of the attributes (for example, number, char, date).

> **Important:** You cannot edit the data type after you save the attribute.

- Set up the Unit of Measure Classes (for example, Currency) and Units of Measure (for example, US Dollars).

- Establish the validation rules for each attribute and create the corresponding value sets. For more details, see Defining Value Sets for User-Defined Attributes.

- Decide how you want the attribute group to be displayed:

  - Single row Displays a Text field with a value.

  - Multi-row Displays multiple values (rows) for the same attributes (column) in a table.

    > **Important:** Define which attributes or combination of attributes will maintain uniqueness of records in cases where the attribute group is displayed as multi-row.

The figure below shows how the display options are set for the different attributes. The figure on Defining Change Attribute Types shows the results of these display option settings. For example, you can see that the attribute Customer is set to display as a text field, and indeed it does. Note that the attribute Customer uses the value set Customer, which is really a table that is displayed as a list of values (LOV) text field. Also notice that the attribute Customer is maintained as part of a unique key. You can add to/edit the unique key as long as doing so does not destroy the uniqueness (creating duplicates) of existing records.

Also note in the figure below that Enabled parameter is set to No for the Company Name attribute, indicating that this attribute will not be displayed on the change page. You can always disable attributes; however, you cannot delete attributes if the attribute group has already been associated with a change type.

Defining change attributes and attribute groups

Defining change type attributes (setting attribute values)

See Also: For more details about creating attribute groups and attributes, see "Defining Item Attributes and Attribute Groups" and the APC online Help (iHelp).

Defining Header Types

Change types provide enterprises a way to accommodate business processes and classify types of changes within a change category. For example, you can define different issue types such as Quality Issue, Product Issue, and Performance Issue to capture the various types of issues in your enterprise.

You can configure a change type to have the following:

- **Auto Tracking Numbers**

  You can specify a prefix and a next available number of a sequence. For example, an Engineering Design Change type can be configured to have the prefix DSGN.

- **Default Assignment Role**

  You can specify an item role (such as Design Engineer) as the default assignment role for a specific person or a group. For example, if Design Engineer is selected, then the person or group who has the Design Engineer role on the subject item for which the change is created is assigned to the change. Note that the change is only assigned to one person or group, thereby ensuring that it is assigned only to the person or group for which it was intended. If more than one person must have the specified role, it is recommended that you create a group and place all users with the same role into the group, and then give that group the default assignment. If more than

one person has the Design Engineer role, then the change would be randomly assigned to one of those users.

- **Attribute Groups/Pages**

  You can associate attribute groups that have been defined for headers to a change type. These attribute groups enable you to capture additional information regarding the type of change being created, or business process-specific attributes required to process the change. You can create pages for a change type and then organize the attribute groups associated with the change type in a logical manner.

- **Setting up valid codes**

  You can specify the valid codes for Priority, Reason and Classifications for the Change type. (Classification codes are only available change types based on the Change Order category.) This allows you to limit the available values for a user to only the selected values for each of these codes. The creation of these values for the different codes was detailed earlier in this chapter.

- **Configuring Primary Attribute groups and sections**

  Depending on your business process, you may elect to enable only certain primary attributes and sections for a change type. For example a Marketing Issue, you can elect not to enable the Project Name and Project Task Name attributes, as these would most likely be not used. Similarly for a Supplier Requested Issue, you can disable the People section, as you may not want the Supplier to have a visibility on who has what role in your organization.

- **Workflow**

  You can define the various statuses for the change type under the workflow tab. Associated with each status you can specify the valid statues for promotion and demotion as well associate a workflow with each status. For an Approval status only an Approval type workflow can be selected. Also if desired you can enable Digital Signatures for the Approval. If a workflow is associated with a status, you also have the option to enable Auto-promote and Auto-demote. This will allow the change to be automatically promoted to the selected status on the successful completion of the workflow or the automatic demotion to the selected status on the rejection of a workflow.

- **Organizational Policies and Task Templates for Change Orders**

  For change orders only, you can specify task templates and propagation rules by organization. You can group a set of change tasks (defined in a task template) to be performed during or between certain statuses of the change type's workflow. You can also specify whether or not all or some of these tasks are mandatory.

Associating Task Templates per organization

Organization Policies

(Cancel) (Apply)

Change Category **Change Order**   Type **ECO**

Organization

• Indicates required field

Select •Organization	Organization Name	Delete
☉ V1	Vision Operations	🗑
○ M1	Seattle Manufacturing	🗑

(Add Another Row)

Organization : V1

Task Templates	Propagation Rules

Task Template	Start After Status	Complete Before Status	Delete
CO Analysis	[▾]	Approval ▾	🗑
CO Release	Approval ▾	Released ▾	🗑
CO Schedule Global	Approval ▾	Scheduled ▾	🗑
CO Implementation Global	Released ▾	Implemented ▾	🗑

(Add Another Row)

Task Templates	Propagation Rules

(Cancel) (Apply)

- **Propagation Rules for Change Orders**

  Besides associating Task templates for each organization, you can specify the Propagation rules for propagating change orders of this type to other organizations. A propagation rule enables you to specify an organizational hierarchy to propagate the change order, including all associated revised items and tasks. The propagation rule can be executed automatically when the change order enters a specific status.

Defining Propagation Rules

Defining Line Types

Change line types enable you to capture specific changes to an item or tasks related to a change. For example, you can define different line types to capture item related changes (such as Item Attribute changes, Component redesign changes, Document changes and Part Obsolescence).

Change Line Types

Creation of a Line type is similar to that of the Change Header type. Each Line type could have a default assignee as well as an associated Subject.

Defining a Change Line Type

Create Change Request Line Type

* Indicates required field   (Cancel) (Apply)

* Name [   ]
Description [   ]

* Start Date [01-Apr-2005]
(example: 17-Mar-2005)
End Date [   ]
(example: 17-Mar-2005)

Subject Selection Criteria
* Subject [Item ▼]

Select	*Attribute	*Value	Allowed
	No data exists.		

(Add Another Row)

Default Assigned To
Assignee Type [▼]   Assignee [   ]

(Cancel) (Apply)

Associating Change Type Attributes

After creating the attributes, values, value sets, and attribute groups, associate the attribute group to a change type or a line type. You can define pages to display the associated attribute groups.

Note: You can reuse the same attribute group across change types belonging to different change categories.

To associate attribute groups with a change type:

1. Add the attribute group to the Attribute Groups list for that change type (change order or issue).

    For example, the change category Change Order has the change type ECO. The following figure shows that the attribute groups Implementation Cost and Unit Cost are associated with the header type ECO.

    Associate change type attribute

    Attribute Groups

    Change Category **Change Order**   Header Type **ECO**

    (Add Attribute)

    Select Attribute Groups and...   (Delete)
    Select All | Select None

Select Display Name	Description	Data Level	Classification	Pages	Update Actions
☐ Unit Costs		Change	ECO	Cost Information	✎
☐ Implementation Costs		Change	ECO	Cost Information	✎

2. Define change pages for the associated attribute groups. You can locate one or more attribute groups on a page, or create separate pages for each attribute group. The following figure shows the change order type ECO has two pages: Classification and Impact Analysis. The figure "Create pages for attributes" shows an example of how two attribute groups are rendered on a single page for a change order.

Create pages for attributes

Change page for attributes on change order overview page

Associating Change Line Type Attributes

You can configure a line type to have Attribute Groups/Pages. You can associate attribute groups that have been defined for lines to a change type. These attribute groups enable you to capture additional information regarding the type of change being created, or business process-specific attributes that are required to process the change. You can create pages for a line type, and then organize the attribute groups associated with the line type in a logical manner.

Attribute Groups for a Change Line Type

Setting Up Change Attribute Group Security

When implementing role-based change management security you can set up privileges to control the view and edit permissions for specific change management attribute groups. You can control which users can view and/or edit certain attribute groups for a change object by assigning a role granting those specific privileges. By default, a change role's View Basic Change Information and Edit/Delete Change privileges control whether you can view or edit attributes that are not controlled specifically at the attribute group level. In other words, when implementing change management security you do not have to specify a view or edit privilege for each attribute group.

Suppose your company is co-designing with your supplier a new motherboard for its next generation of desktop computers. To improve communication with your supplier on design changes to the motherboard you would like to securely share change request information externally with your suppliers and contract manufacturers. The Supplier Engineer should only be able to view specific attribute groups.

To implement attribute group security for change management:

1. Select the Application Developer responsibility, navigate to the Form Functions form, and create Form Functions for each privilege that controls view and edit permissions for the attribute groups. Specify "Change" or "Change Line" in the Object field for each Form Function.

2. Select the Development Manager responsibility and navigate to the Setup Workbench. On the **Attribute Group Details** page for each attribute group, specify the View Privilege and Edit Privilege in the Data Security section.

3. On the **Change Role Detail** page, grant the privileges that you defined as Form Functions in the previous steps.

   See Also: For more details about setting up attribute group security for Change Management, see "Setting Up Item Attribute Group Security" and the following topics in the APC online Help (iHelp):

   • Creating Custom Privileges

   • User Defined Attributes

   • Managing Roles

Implementing User Defined Functions for Change Type Attribute

You can define your own custom logic by adding user-defined functions and actions to existing change pages. In such cases you do not need to customize the entire page.

You can register user-defined functions for executing customer-specific business rules and calculations. These functions can be written in Java or PL/SQL. URL functions can also pass the values of certain parameters to the URL string and redirect users to a secure page. For each function, you need to register the list of parameters, their data types, and how they get their values by specifying parameter types.

Actions are trigger points for functions and can be displayed as buttons or links. You can determine the conditional visibility of the button/links and also prompt the user based on his or her input.

Example: Implement User-Defined Functions for Change Type Attributes

The following example shows how you would calculate Total Cost by using the attribute group Implementation Cost.

Sequence	Attribute Group: Implement Cost	Attribute Group: Implement Cost	Attribute Group: Implement Cost	PL/SQL Function: Calculate Cost	PL/SQL Function: Calculate Cost	PL/SQL Function: Calculate Cost
Sequence	Attribute Name	Data Type	Mapping Attribute & Parameters	Parameter Name	Data Type	Parameter Type
1	Suppliers Charges	Number	--à ---à ---à	Cost 1	Number	Input
2	Manufactur-ing	Number	--à ---à ---à	Cost 2	Number	Input
3	Engineering	Number	--à ---à ---à	Cost 3	Number	Input
4	Field/Repair	Number	--à ---à ---à	Cost 4	Number	Input
5	Total Implement Cost	Number	--à ---à ---à	Cost Result	Number	Return Value

First, register the PL/SQL function "Calculate Cost" (based on the PL/SQL procedure that resides in the PL/SQL package) with all required parameters.

Then navigate to the "ECO" change order type and add the action to the attribute group "Implement Cost" by clicking Update Action. The action executes the function on the change pages. After creating the action, map the function parameters to the corresponding attributes.

The mapping section on the **Action Detail** page provides the mapping information for function parameters. You can also map parameters of functions to the object's primary key value (for example, CHANGE_ID for change objects).

When setting up an action:

1. Specify a role-based privilege to secure the action.

2. Add the Dynamic Visibility Function (written in Java only) to your action if there is a certain condition that needs to be satisfied before users can view the action button or link. For example, if one or all of the fields are empty, then the user cannot see the button/link to execute the action.

3. Add the Dynamic Prompt Function (written in Java only) to your action if the label for the button/link needs to be changed depending on a certain condition. For example, it the Total Implement Cost attribute is empty, then the button label is Apply, otherwise the button label is Apply Changes.

> **Important:** Dynamic Prompt and Dynamic Visibility function parameters must be mapped to the corresponding attributes as well.

Associate action with attribute group to execute a function

Action Details for Calculate Cost

Basic Information

Edit

Object Name **Change**
Classification **ECO**
Attribute Group **Implementation Costs**
Sequence **1**
Action Name **Calculate Cost**
Description
Function **Calculate Cost**
Security Privilege

Execution Method **User Action**
Display Style **Button**
Prompt Application **Oracle Engineering Online**
Prompt Message Name **Calculate Cost**
Visible **Yes**

Dynamic Prompt Function
Dynamic Visibility Function

Mappings for Function Calculate Cost

Edit

Parameter Name	Mapped To	Mapped Attribute
Cost 1	Attribute Group	Suppliers Charges
Cost 2	Attribute Group	Manufacturing
Cost 3	Attribute Group	Engineering
Cost 4	Attribute Group	Field / Repair
Cost Result	Attribute Group	Total Implement Cost

Change type attribute page displays action button

Implementation Costs

Calculate Cost

* Indicates required field

Material	137.5	Dollars	Suppliers Charges	175	Dollars
	Costs related to material scrap plus rework			Costs related to supplier rework, scrap, tooling, cancel fees, and return credits	
Manufacturing	150.65	Dollars	Engineering	120.87	Dollars
	Costs related to manufacturing tooling, test equipment, facilities			Costs related to engineering design and drafting	
Field / Repair	250.45	Dollars	Quality Assurance	160	Dollars
	Costs related to field service and repair labor, rework, and scrap			Costs related to quality assurance	
Total Implement Cost	994.47	Dollars			
	Result				

Cancel Apply

Related Topics

See Also: For details about user-defined functions, see "Implementing User-Defined Functions for Items" and the APC online Help (iHelp).

Defining Change Category Search Criteria

Criteria templates offer you a way to save frequently used search criteria—essentially, acting as a saved list of specific attributes and attribute values. Criteria templates that you—the administrator—create are available to all users, and serve to expedite searches for issues, change requests and change orders, thereby saving users the time it would take to specify search criteria and execute frequent change management searches.

Of course, users also have the option of creating criteria templates for their own private use. In fact, when both a user and administrator define a default criteria template for a particular change category, the user-defined criteria template takes precedence.

Because criteria templates contain user-defined attributes, you must always define them in the context of a change category. You can define as many criteria templates as you wish, and you can also denote the most commonly used as the default criteria template. For example, you may define criteria templates to quickly find Open Change Requests, My Issues, or Scheduled Change Orders.

See Also: For more details about creating change category criteria templates see the APC online Help (iHelp).

Defining Change Category Display Formats

Results formats enable you to predefine search results views for each change category. You can use these views to look at different sets of attributes of the change objects (for example, issues, change requests, change orders) that are returned by a search. Both administrators and users can create results formats. Administrator-created results formats are available to all users. User-created result formats are available only to the users who created them. You can define as many results formats as you wish, and you can also denote the most commonly used as the default results format.

You can include any base or user-defined attributes in your results format definitions. However, multi-row attribute groups cannot be displayed using results formats. You can include display sections in your results formats that provide links directly to the section from your search results. This enables you to display links in your search results to any user-defined change type page (such as Cost Information) or any of the standard display sections: Lines, Attachments, Action Log, Approval, Approval History, Dependencies, Revisions, and People.

See Also: For more details about creating change category results formats, see the APC online Help (iHelp).

Change request search results using Cost Information results format

Defining Change Category Reports

Change management reports are basically search criteria and display formats that you can save, browse, email, or print. You can create reports for any change management category. You can also browse a report sequentially or using a summary view. Browsing a report sequentially enables you to step through each change object in a report. A summary view displays the report in tabular column format. You can send a report to other users including registered suppliers and customers.

An Admin can create a report for the most common or frequent searches that users of the system would be performing. This would be based on the business process being implemented. These Admin defined reports would save an ordinary user the time

of selecting a change category, search criteria and display format, and repeating the same search over and over again. You can give reports meaningful names as well. For example, a search for all open high priority issues may yield dozens of issues, each identified by issue name and number. You can name the report for these search results " High Priority Open Issues Report" as shown in the figure below.

Report security is consistent with search security: you can only access change object on which you have the required role. Also, administrator-created reports are available to all users, although only the administrator can edit these reports.

High Priority Issue Report

6

Defining Item Structures

Overview of Item Structures

Use the following tasks to define item structures.

Task	Required?
Defining Structure Types	Yes
Defining Structure Names	Yes
Defining Item Attachment Categories for a Structure Type	
Defining Criteria Templates for a Structure Type	
Defining Results Format for a Structure Type	

Defining Structures

Defining Structure Types

A structure type enables you to classify and characterize item structures throughout the lifecycle of an item. For example, you can have structure types for Engineering, Manufacturing, and Service. You can define an unlimited number of structure names for a structure type, enabling you to create several different structures of the same type that you can use for comparison. For example, you can create several different engineering BOMs (EBOM1, EBOM2, EBOM3) using the structure type Engineering BOM with different component costs and BOM grading attributes. Comparing these BOMs enables you to make better decisions earlier in the lifecycle of your products.

You can associate item attachment categories, criteria templates and results formats with a structure type. Structure types are hierarchical, with structure names, item attachments categories, criteria templates and results formats inherited throughout the hierarchy.

Defining Structure Names

You can define multiple structure names for each structure type and enable them in different organizations. For example, you can have three engineering BOMs (EBOM1, EBOM2, EBOM3) that use the structure type Engineering BOM; you can use these to compare designs. Manufacturing BOMs may be slightly different in each global region because similar parts are procured from different suppliers. You

can have a structure type Manufacturing BOM with different structure names (MBOM_US, MBOM_EMEA, MBOM_APAC) for each of your manufacturing regions around the globe. Structures within a structure type share the same common characteristics.

Defining Item Attachment Categories for a Structure Type

Associating attachment categories with a structure type enables you to view item attachments to structure components within a BOM. Suppose you have a CAD drawing attached to each of the major subassemblies in each of your engineering BOMs. You can associate the attachment category CAD Drawing to the structure type Engineering BOM. The CAD drawings for each of the subassemblies will then appear in the EBOM structures.

Defining a Criteria Template for a Structure Type

Criteria templates enable you to save frequently performed search criteria based on a list of attributes. Criteria templates for structures enable you to search components within a structure. You can associate with a structure type an existing criteria template for a given item catalog category, or create a new criteria template to search items within a structure.

Administrator-created criteria templates are available to all users. This helps users quickly find components within a structure without having to create their own criteria templates. You can specify a default criteria template for each structure type.

Defining Results Format for a Structure Type

Results formats enable you to predefine search results views for a structure type. You can use these views to look at different sets of attributes of component items within a structure. Both administrators and users can create results formats. Administrator-created results formats are available to all users. User-created result formats are available only to the users who created them. You can define as many results formats as you wish, and you can also denote the most commonly used as the default results format.

You cannot display multi-row attribute groups using results formats.

You can define a results format by listing the common information a typical user might wish to view for components in a structure. For example, you might select the attributes Item Name, Description, Component Quantity, Capacitance Value, or Tolerance. Results formats for structures enable you to include navigation links for actions such as item Where Used, First-Level components, and View Components in Hierarchy.

See Also: For more details about managing results formats see the APC online Help (iHelp).

Display Formats and Search Criteria for Structures

Search Criteria

Administrators can define search criteria for a given structure type. All search criteria defined by administrators are available to all users viewing a structure of the given structure type to which the search criteria is associated. On selecting a search criteria, only those components that satisfy the criteria will be listed in the hierarchy and

the components that do not match the criteria will be filtered out. Intermediate nodes will be listed even though they might not satisfy the criteria to maintain the context for a lower level component. Primarily search criteria are useful to filter only relevant components in a bill having a large number of components. Search criteria templates can be built using item and component base and user defined attributes.

Display Formats

Administrators can define display formats for a given structure type. All display formats defined by administrators are available to all users viewing a structure of the given structure type to which the display format is associated. Display formats allow user to view attribute information across an entire structure. The display formats can include item and component base attributes and user defined attributes.

Oracle PLM provides the following system defined display formats:

Display Format Name	Fields Included in the Display Format
Component Information	Component Name, Description, Revision, Category
Component Details	Component Name, Description, Revision, Item Sequence, Operation Sequence, UOM, Quantity, Planning%, Yield, Item type
Change Order	Component Name, Description, Revision, Implemented, Change Order
Comments	Component Name, Description, Revision, Comments
Effectivity	Component Name, Description, Revision, Effectivity Control, Effective From, Effective To, Change Order
Lifecycle Status	Component Name, Description, Revision, Lifecycle Phase, Approval Status, Item Status, Percent Complete, Schedule End Date, Progress Status
Long Description	Component Name, Description, Revision, Long Description
Material Control	Component Name, Description, Revision, Supply Type, Subinventory, Locator, Auto Request Material, Enforce Integer Requirements
Order Management	Component Name, Description, Revision, Check ATP, Optional, Mutually Exclusive, Minimum Quantity, Maximum Quantity, Basis
Reference Designators	Component Name, Description, Revision, Quantity, Quantity Related, Reference Designator
Shipping	Component Name, Description, Revision, Shippable, Include on Shipping documents, Required to Ship, Required for Revenue

User Defined Attributes for Structures and Components

User defined attributes for structures and/or components allow the user to capture specific details about the structure or component. The attributes whose value(s) can be captured for a Structure, are defined by the administrator, while defining the Structure Types. User defined attributes has an administrative workbench which helps the setup of the attribute groups, attributes and the value set tied to an attribute, if any. Different types of structures and its components will have different attribution requirements, For example, when performing analysis of various design scenarios, every structure may need to capture the weight and cost attributes. Similarly, components may need to capture attributes like "Mean Time between failure" which are unique to a component usage in a specific structure.

When setting up the Structure Types, administrators can setup the attribute groups associated with that Structure Type.

Administrative Setup

Attribute Groups List for Structure Types and Component Types

Administrators can add attribute groups and define the association level of the attribute group, which can be either the Structure or the component.

Attribute Group Definition for Structure and Component Types

End User View

The end user view of extensible attributes would be based on the administrative definition of the attribute groups/attributes and the value set tied to them. The pages would be generated dynamically.

Sample end user view for Engine Specifications:

User View of an Extensible Attribute Group

Engine Specifications
Item **M10000** Revision **B**

Horse Power	150
Shaft Length/Inches	L20
Weight/Lbs	L461
Cylinders	V-6(60)
Displacement	164.3(2693)
Bore x Stroke	3.31 x 3.19 (84 x 81)
Operating Range/RPM	5200-6000
Starting System	Electric Start(w/ Suzuki EFI)
Oil Tank Capacity	9.0(8.5)
Lubrication	Oil Injection
Ignition	Micro Link
Alternator	12V 420W
Piston Ring Type	Keystone
Mounting	Shear Mount
Trim Position	-4 to +20
Fuel	Unleaded 91 Octane

Valid Component Type Rules

The system enables you to add valid component user items to a structure. It also enables you to copy a structure to a new structure when the item list of values includes only valid items that have defined component types for the parent item type.

Adding Components to a Structure

When adding components to a structure using the 'Add' right click action, the Item Search and Item Advanced Search restricts the list of component items that can be added as components to an item. This list will include only the valid component user items configured for the item type of the parent/subassembly.

When adding components to a structure by dragging and dropping components from Favorites, Item Search Results, the system restricts the list of component items that can be added as components to an item. The component items, which can be dragged and dropped, include only the valid component user items configured for the item type of the parent/subassembly.

Structure Copy

When copying a structure to a new structure, the Item list of values (LOV) for the target structure includes only the valid items that have defined component types for the parent item type. Therefore, when you are defining the valid component item types for parent item types, if the parent item type is also a valid component item type, define it as a component item type, as well. Complete the same validation when copying a structure for a subassembly , which is added by dragging from another structure window.

Structure Bulkload and Open Interface

When importing a structure through bulkload or the open interface, the system automatically considers item type validations.

Functional Rules

- Define User Item Types in the Item Master

- Define Valid Component Item Types for a Parent User Item Type

- Validate a parent item type with multiple component types in an OR condition

- Item LOV's in Add Component and Item Advanced Search will validate the valid component item types for a parent item type to which the component is being added

- Target Structure Item LOV will validate the valid component item types for a parent item type

- Structures created through bulkload or Open interface need to validate the valid component item types

- Once structures have been defined with existing valid component types on deletion of a component user item type for a parent item type, all new structures will honor the new component user item types definition.

7

Setting Up Document Management

Overview of Document Management

Oracle's E-Business Suite provides integration to Oracle Files, providing users a wide range of document management capabilities. Oracle Files provides a great deal of flexibility for organizing unstructured data. This step identifies some options you should consider when organizing files and folders in an Oracle Files repository.

Oracle PLM Family Pack C includes Document Management functionality which is integrated with Oracle Collaboration Suite Release 2 (9.0.4). It allows the end user to build enterprise content management repository for E-Business Suite and integrate documents into business processes. If you would like to use this functionality, you will need to install and configure Oracle Collaboration Suite Release 2 (9.0.4). See Implementing Oracle Files with Oracle E-Business Suite 11i, note 284077.1 on OracleMetaLink, for instructions on how to install and configure Oracle Collaboration Suite with Oracle Applications.

Working with Oracle Files

Oracle Files is an innovative content management system designed to support user collaboration. Through its easy-to-use web interface and robust content management features, Oracle Files makes it simple to access, manage, and share data with any Oracle Files user.

Managing Workspaces or Shared Folders

Oracle Files introduces the concept of *workspaces or shared folders*. Workspaces are folders that are accessible by a defined set of users who are assigned one of three sets of privileges:

Administrator

The role of the workspace Administrator is to manage the workspace. The management role includes such functions as adding and removing workspace members, modifying member access, requesting additional workspace quota, and specifying the visibility and workspace type. The workspace Administrator has full access to the files and folders in the workspace, and can unlock files that have been locked by other workspace members.

Participant

A workspace Participant can edit the contents and properties of files and folders in a workspace. A workspace Participant can copy and move items to and from the

workspace, and can delete files and folders from the workspace and the workspace's Trash folder.

Viewer

A workspace Viewer is restricted to viewing workspace files and folders.

Managing Folders

In addition to workspaces, Oracle Files provides for a further logical grouping of files into folders. You can create a folder where you have the appropriate access level. You can create folders in your Private and Public folders, as well as in any workspace in which you are a Participant or Administrator.

You can access files and folders in your My Private Files, My Public Files, My Workspaces, and Trash folders. Each folder confers a different level of access.

My Private Files

Each user has a My Private Files folder. Only you (the owner of a My Private Files folder) can access your folder. You have complete control over its contents.

Note: The Administrator also has full access to each user's My Private Files folder. For more information about private files, see the online Help for Oracle Files.

My Public Files

Each user has a My Public Files folder. You have complete control over the contents of this folder, and no one beside you or the Administrator can add or remove anything from it. This folder, however, is visible to all users, and all users can copy the contents of your public folder, or send a link to any of its files or folders.

All Public Files

All Oracle Files users can view all the files in All Public Files. From All Public Files, you can view the contents of the enabled Public folders of Users and the enabled Public folders of Workspaces.

My Workspaces or Shared Folders

Access to files and folders in My Workspaces and Workspace Trash is exclusive to the members of each respective workspace. Users who are not members of a particular workspace cannot access its files and folders. However, users can access the contents of the workspace's Public folder. Members' access levels determine the actions workspace members can perform on the files and folders in a workspace.

Deleting Files and Folders

Oracle Files has two types of Trash:

Personal Trash

When a user deletes a file or folder from My Private Files or My Public Files, the files or folders move to Trash. Once there, you cannot modify the item or its properties. You can, however, move, delete, and empty the contents of Personal Trash. Like My Private Files, you are the only user in the Subscriber who can access your Personal Trash.

Workspace Trash

When you delete a file or folder from a workspace, the file or folder moves to Workspace Trash. Once in Workspace Trash, you cannot modify the properties of that file or folder. Only Participants and Administrators can move, delete, and empty the contents of Workspace Trash. However all workspace members can copy a file or folder out of Trash.

Structuring Folders/Workspaces

As previously mentioned, Oracle Files provides a great deal of flexibility in the structuring of data. The following security considerations should factor in to implementation decisions about the organization of unstructured data (documents/ files/ folders):

- Examine what content is internal to the enterprise and what may be shared externally as part of the collaboration process or on a website.

- Access privileges by organizations modeled as Organizations in EBS.

- Access controlled by item catalog categories or types of items.

- Access controlled by business objects in EBS such as Items, Change Management, Projects, and Purchase Orders.

- Access controlled by the lifecycle state of the item.

Following are some examples for structuring folders/workspaces.

Example 1: Model Workspaces as Item Catalog Categories

/Engines	Workspace	Item Catalog Category
/M10000	Folder	Item
/M10000 Rev 1.0	Sub Folder	Item Revision
/Functional Specification	Sub Folder	Doc. Type
/CAD View	Sub Folder	Doc Type
/M10000 Rev 2.0	Sub Folder	Item Revision
/M10000 CO	Folder	Item specific CR/CO
/Manufacturer 1	Workspace	Manufacturer Name 1
/Manufacturer 2	Workspace	Manufacturer Name 2
/Change Management	Workspace	Generic CR/ CO
/Intranet	Workspace	Public to Enterprise
/Internet	Workspace	Public to www

Example 2: Model Workspaces as Organizations or Divisions in a Company

/Org 1	Workspace	Organization 1
/M10000	Folder	Item
/M10000 Rev 1.0	Sub Folder	Item Revision
/Functional Specification	Sub Folder	Doc. Type
/ CAD View	Sub Folder	Doc Type
/M10000 Rev 2.0	Sub Folder	Item Revision
/M10000 CO	Folder	Item specific CR/CO
/Org 2	Workspace	Organization 2
/Manufacturer 1	Workspace	Manufacturer Name 1
/Manufacturer 2	Workspace	Manufacturer Name 2
/Change Management	Workspace	Generic CR/ CO
/Intranet	Workspace	Public to Enterprise
/Internet	Workspace	Public to www

Example 3: Model Workspaces as Business Objects in EBS

/Items	Workspace	Business Object
/Engine	Folder	Item Catalog Category
/M10000	Sub Folder	Item
/M10000 Rev 1.0	Sub Folder	Item Revision
/Functional Specification	Sub Folder	Doc. Type
/ CAD View	Sub Folder	Doc Type
/M10000 Rev 2.0	Sub Folder	Item Revision
/M10000 CO	Sub Folder	Item specific CR/CO
/Change Management	Workspace	Generic CR/ CO
/Project Management	Workspace	Business Object
/Manufacturer 1	Workspace	Manufacturer Name 1
/Manufacturer 2	Workspace	Manufacturer Name 2
/Intranet	Workspace	Public to Enterprise
/Internet	Workspace	Public to www

Example 4: Model Workspaces as Lifecycle Phases

/Concept	Workspace	Bus. Object
/Engine	Folder	Item Catalog Category
/M10000	Sub Folder	Item
/M10000 V. A	Sub Folder	Item rev
/Functional Specification	Sub Folder	Doc. Type
/ CAD View	Sub Folder	Doc Type
/M10000 V. B	Sub Folder	Item rev
/Design	Workspace	Generic CR/ CO
/Test	Workspace	
/Release to Mfg	Workspace	
/Engine	Folder	Item Catalog Category
/M10000	Sub Folder	Item
/M10000 Rev 1.0	Sub Folder	Item rev
/Functional Specification	Sub Folder	Doc. Type
/ CAD View	Sub Folder	Doc Type
/M10000 Rev 2.0	Sub Folder	Item rev
/M10000 CO	Sub Folder	Item specific CR/CO
/Obsolete	Workspace	
/Manufacturer 1	Workspace	Manufacturer Name 1
/Manufacturer 2	Workspace	Manufacturer Name 2
/Intranet	Workspace	Public to Enterprise
/Internet	Workspace	Public to www

For more details about managing attachments using Document Management, see the Oracle Applications System Administrator's Guide.

Document Review and Approval

Document Attachment Review or Approval provides:

- Flexible and scalable way of defining different types of reviews/approvals;

- Capture of user-defined attributes for them;

- Secure collaboration

- Flexible and advanced approval routing definition.

Complex parallel and serial approval routes can be created as predefined templates to enforce strict approval processes, or ad hoc routings to obtain additional approvals within the enterprise can be specified.

Attachments could be selected and submitted for Review or Approval. The Review and Approval workflows maybe independent of each other. The Approval and Review process will use predefined approval templates with support for ad-hoc addition of new steps. You can add steps to the process only if a workflow routing is associated with the particular status. You cannot delete predefined (in the item catalog category) steps. You can add additional assignees to a step, but you cannot delete predefined assignees. Based on the Approval or Review type chosen, the workflow/approval routing is determined. Appropriate notifications are sent at the various stages of the approval process including a final notification of approval. Approvers could be derived from user's role on the business object.

The system indicates the current status of the document. A document could have one of the following statuses of Approved/ Reviewed / Submitted for Approval/Submitted for Review or Rejected. If the Document status is not one of these, then it means that the document has not been submitted through a formal review and approval process and is in a status of Draft or Unapproved. As the document progresses through the Approval process the document status should be automatically updated and the approval or review itself should be appropriately routed to different users for their review or approval.

The Document Review and Approval process is built using the Change Management Framework with Approval and Review being modeled as Change Categories. Thus, creating the different Approval and Review types is similar to Defining the Header Types for a Change Category, with a few minor differences. For example Lines and associated Line types cannot be defined for the Approval and Review types. Similarly the References, Dependencies and Attachments sections are not available.

Once a Document Approval or Review type has been defined, then with each status a workflow could be associated. An Approval type must have an Approval status with an approval workflow associated with it. For more details about creating workflows see Defining Workflow Templates. To setup and use "Document Approval and Review" you must have installed and configured Oracle Files with Oracle Applications (see About doc for PLM FP-C metalink note 242348.1)

Change Management for Documents

Change Management for documents provides change control functionality for all types of attachments a crucial and important piece of non-structured information about the product.

Types of attachment changes supported using a change order:

- adding attachments

- deleting attachments

Change control is achieved by providing mechanism to author, track and manage changes to any of the entities above through a change order approval and implementation process. It is possible to:

- Specify items and documents that will be revised or effected in a change order

- Author and specify the exact changes that are desired or required to resolve an issue or change request using an easy to use UI

- Route the change order for approval to the appropriate people responsible for the revised item based on item roles, change desired and lifecycle of item.

Once the change order is approved the changes should be implemented manually or automatically on the specified implementation date.

Change Management for Documents is supported through Change Management functionality available in APC. User can create a Change Order and specify changes to the attachments of a reviewed item. There are no additional steps required in the setting up of a change order to be able to do attachment changes for a revised item other than those mentioned in the step Defining Change Categories and Types.

However, to setup and use "Change Management for Documents" you must have installed and configured Oracle Files with Oracle Applications (see About doc for PLM FP-C metalink note 242348.1)

8

Defining Roles

Overview of Roles

APC provides role-based security to specify the privileges assigned to each user on an item, catalog, catalog category, and change category. A *role* consists of a set of privileges granted to a user on a specific object, such as an item. For example, you may assign the Design Engineer role to some of the engineers within your organization for a set of items. The Design Engineer role grants them the privilege to view and edit the item. To simplify the maintenance of role-based security, you can define groups of users. You can assign to an item or set of items (for example, all items in an item catalog category) a role for a person, group, company, or all users.

Task	Required?
Defining Change Roles	Yes
Defining Item Roles	Yes
Defining Item Catalog Category Roles	
Defining Catalog Roles	

Defining Roles

Defining Change Roles

The change role assigned to a user for a change object (for example, issue, change request, change order) determines which actions that user can perform on the change object. For example, a user with an Approver role on a change request is granted the View Basic Change Information and Edit/Delete Change privileges. You can also specify which user-defined attribute groups a user can view and/or edit when granted a change role (see "Setting Up Change Attribute Group Security").

You can assign change roles to a person, group, company, or all users. To simplify maintaining change management security, you can assign change roles directly to the change object or inherit them through item role mapping from the subject item of the change header or change lines. For example, you can map the Design Engineer item role to the Change Design Engineer role for issues, change requests, and change orders. So users with the Design Engineer role on the subject item of the change request header or lines will inherit the Change Design Engineer role as well. You can also assign a default role to all internal users with the site level profile ENG: Internal User Default Role for

Changes. A role that is explicitly granted to a user for a change object is a *direct role assignment*. Roles inherited from an item are *inherited role assignments*.

Following are the seeded change roles in Oracle APC:

- Approver

- Assignee

- Creator

- Requestor

- Reviewer

See Also: For more details on creating change roles, see "People and Roles Administration" in the APC online Help (iHelp).

Item role to change role mapping

Item Role Detail

Name **Design Engineer**
Description **PLM Design Engineer**
Object Name **Item**

Item Privileges

Name	Description
View Target Markets	View Target Markets
Add/Delete Item Attachment	Add/Delete Item Attachment
Add/Delete Item Project/Task Association	Add/Delete Item Project/Task Association
⋮	⋮
View Item/Item Revision LifeCycle	View Item/Item Revision LifeCycle
View Key Metrics	View Key Metrics

Role Mappings

(Edit)

Type	Role Name
Issue	Change Design Engineer
Change Order	Change Design Engineer
Change Request	Change Design Engineer

Defining Item Roles

The item role assigned to a user for an item determines which actions that user can perform on the item. For example, a user with the Design Engineer role is granted the View Item and Edit Item privileges, among other privileges.

You can assign item roles to a person, group, company, or all users. To simplify maintaining item security, you can assign item roles at the item, item catalog category, or organization level. For example, you can assign a user the Design Engineer role for the Motherboard item catalog category; this in turn assigns the Design Engineer role to the user for all Motherboard items. A role that is explicitly granted to a user for an item is a *direct role assignment*. Roles assigned at the organization or item catalog category level are *inherited role assignments*. You can map an item role to a change role for each

change category. For example, you can map the item Design Engineer role to the Change Design Engineer role for issues, change requests, and change orders. So users with the Design Engineer role on the subject of the change request header or lines will inherit the Change Design Engineer role as well.

Following are the item roles seeded in Oracle APC:

• Item Author

• Design Engineer

• Design Reviewer

• Supplier Engineer

See Also: For more details on creating item roles, see "People and Roles Administration" in the APC online Help (iHelp).

Defining item roles

Defining Item Catalog Category Roles

The item catalog category role assigned to a user determines who can create items in a specific item catalog category. For example, a user with the Catalog Category User role on the Motherboard item catalog category is granted the "Create Item of this Catalog Category" privilege, which enables him or her to create Motherboard items.

The item catalog category role "Catalog Category User" is seeded in Oracle APC.

See Also: For more details on creating item catalog category roles, see "People and Roles Administration" in the APC online Help (iHelp).

Defining Catalog Roles

The catalog role assigned to a user determines who can view and assign items to a catalog. For example, a user with the Catalog Manager role on the Purchasing Catalog is granted the View Catalog and Manage Catalog privileges. The user can view the items in the Purchasing Catalog and add items to the Purchasing Catalog.

Following are the seeded catalog roles in Oracle APC:

- Catalog Manager
- Catalog Viewer

See Also: For more details on creating catalog roles, see "People and Roles Administration" in the APC online Help (iHelp).

9

Implementing Role Based Security

Overview Role Based Security

Task	Required?
Defining Groups	
Implementing Item Catalog Role-Based Security	Yes
Implementing Catalog Role-based Security	Yes
Implementing Change Management Role-based Security	Yes

Defining Role Based Security

Defining Groups

You can create groups of people that include internal users, suppliers and customers. Groups are extremely useful for managing role assignments for items, catalogs, and change management. Groups are also useful in change management approval routings, where one or all people in a group must approve a change, or all people in a group should be notified of a change. You can create groups of people for different teams or departments that serve similar business functions. You can also create groups for different suppliers or customers.

A group consists of one or more members. The person that creates the group is by default a member and the Group Administrator. Each member in a group is assigned a group role that determines which privileges are granted to the member of that group. There three privileges a group role may grant a member: View Group Header, View Group Members, and Manage Group. For example, you can create a Supplier group made up of all your suppliers; however, you do not want any of the members to see who else is in the group. In this case, each of the members would be assigned the Group Reviewer role, which only allows them to see the group header information, but not the other members. In another scenario, you may wish to create an Engineering group where you want members to be aware of who else is in the group. In this case, you can grant each member the View Group Members role. Only a person or group with the Group Administrator role can also manage the group. The Manage Group privilege enables a user to add/delete members and change their roles.

Group details

Implementing Item Catalog Role Based Security

APC provides *role-based data security*, which enables users to secure individual data elements. Every item in APC is secured using role-based data security. Your ability to view, edit and perform certain actions on an item is determined by your role on it. At the item level, roles are assigned to users by anyone who has the privilege to add people. You can assign roles at the organization, item catalog category, or item level.

Privileges

A privilege defines a user's access to an item. For example, a user's ability to view an item, edit it, and promote its lifecycle phase is determined by his or her privileges on that item. You can define additional privileges to secure user-defined attribute groups.

Roles

A role is a collection of privileges. Roles are a convenient way to group privileges into a bundle that can later on be assigned to users, groups, or companies. For example, a user with the Design Engineer role on an item is granted the privilege to View Item, Edit Item, Add/Delete Item Attachments and more. However, the Design Engineer is not granted the privilege to Promote Item/Item Revision Lifecycle Phase.

You can assign item roles to a person, group, company, or all users. To simplify maintaining item security, you can assign item roles at the organization, item catalog category, or item level. You can specify default roles for all items in an organization. For example, you can assign the Item Reviewer role to the Engineering group so that all engineers can search for and view all items in the item catalog. For each item catalog category you can specify which people can create items by assigning them the Catalog Category User role. For example, you can assign just the NPI engineers the Catalog Category User on the item catalog categories for which they are responsible for maintaining. Next, you can assign default role assignments to people by organization for all items in the item catalog category. For example, you can assign your Motherboard

engineers a Design Engineer role for the Motherboard item catalog category. You can assign the buyer for Motherboards a Buyer role. At the item level you can assign direct roles to enable access to specific items. For example, you may assign the role of Supplier Engineer to suppliers with whom you are collaborating on specific items.

Organization level item role assignments

Organization Roles

TIP Default Role Assignments for all objects in the selected organization

Organization Vision Operations   Go
Object Type Items

Update

Select Users and...   Delete

Select All | Select None

Select	Role	Type	Name	Company	Start Date	End Date
☐	Engineering Manager	Person	Steve Williams	Vision Enterprise	04-Aug-2003	
☐	Product Manager	Person	Steve Williams	Vision Enterprise	04-Aug-2003	
☐	Design Engineer	Person	Mary Robinson	Vision Enterprise	04-Aug-2003	
☐	Manufacturing Engineer	Person	Ravi Patel	Vision Enterprise	04-Aug-2003	
☐	Marketing Manager	Person	Cathy Price	Vision Enterprise	04-Aug-2003	
☐	Quality Manager	Person	Sally Martin	Vision Enterprise	04-Aug-2003	
☐	Service Engineer	Person	Jay Shankar	Vision Enterprise	04-Aug-2003	
☐	Safety Engineer	Person	Rebecca Swanson	Vision Enterprise	04-Aug-2003	
☐	Buyer	Person	Catherine Baker	Vision Enterprise	04-Aug-2003	

Item catalog category people

People

Item Catalog Category: **Motherboards**

Item Catalog Category People

TIP These roles apply at the Item Catalog Category level (for example, users with these roles can create an item in this Item Catalog Category).

Update

Select Users and...   Delete

Select All | Select None

Select	Role	Type	Name	Company	Start Date	End Date
☐	Catalog Category User	Person	Steve Williams	Vision Enterprise	31-Jul-2003	

Item People

TIP The following are default User role assignments for all items in this Item Catalog Category in the selected organization.

Organization Vision Operations   Go

Update

Select Users and...   Delete

Select All | Select None   Previous   Next 10

Select	Role	Type	Name	Company	Start Date	End Date
☐	Design Reviewer	Group	Vision Engineering		06-Aug-2003	
☐	Engineering Manager	Person	Steve Williams	Vision Enterprise	06-Aug-2003	
☐	Product Manager	Person	Steve Williams	Vision Enterprise	06-Aug-2003	
☐	Design Engineer	Person	Mary Robinson	Vision Enterprise	06-Aug-2003	
☐	Manufacturing Engineer	Person	Ravi Patel	Vision Enterprise	06-Aug-2003	
☐	Marketing Manager	Person	Cathy Price	Vision Enterprise	06-Aug-2003	
☐	Quality Manager	Person	Sally Martin	Vision Enterprise	06-Aug-2003	
☐	Service Engineer	Person	Jay Shankar	Vision Enterprise	06-Aug-2003	
☐	Safety Engineer	Person	Rebecca Swanson	Vision Enterprise	06-Aug-2003	
☐	Buyer	Person	Catherine Baker	Vision Enterprise	06-Aug-2003	

Previous   Next 10

Select Users and...   Delete

Item-specific role assignments

Implementing Catalog Role Based Security

For each catalog you create you can specify which people can view and manage the catalog. Suppose you've set up a Product catalog, Purchasing catalog, and Service catalog. You can assign the Catalog Manager role to your marketing managers to maintain the Product catalog, while assigning the Catalog Viewer role to All Users so that everyone can view the Product catalog. You can assign your buyers the Catalog Manager role to maintain your Purchasing catalog.

Assigning Catalog Roles

Implementing Change Management Role Based Security

You can set up change management security so that most if not all the role assignments are inherited from the subject item of the change object header and lines, the approval routing, and default assignments. You can also assign a default role to all internal users with the site level profile ENG: Internal User Default Role for Changes.

When defining an item you specify a role mapping for each change category: issue, change request, and change order. For example, suppose a supplier creates a change request for a Motherboard item. Then all people with a Design Engineer role on the Motherboard item will get the Change Design Engineer role on the change request. This is the most efficient way to set up change management security.

When you defined your change types in "Step 4: Defining Change Categories and Types," you specified one or more valid approval routings. Each person in the approval routing receives a role on the change object based on the approval routing activity. All Request Approval assignees get an Approver role. Request Comment and FYI assignees get a Reviewer role on the change object. You can also specify a Default Assigned To role for a person/group when you define a change type. The Assigned To person for every change object gets an Assignee role.

Item role to change role mapping

Item Role Detail

Name **Design Engineer**
Description **PLM Design Engineer**
Object Name **Item**

Item Privileges

Name	Description
View Target Markets	View Target Markets
Add/Delete Item Attachment	Add/Delete Item Attachment
Add/Delete Item Project/Task Association	Add/Delete Item Project/Task Association
⋮	⋮
View Item/Item Revision LifeCycle	View Item/Item Revision LifeCycle
View Key Metrics	View Key Metrics

Role Mappings

(Edit)

Type	Role Name
Issue	Change Design Engineer
Change Order	Change Design Engineer
Change Request	Change Design Engineer

10

Understanding Product Data Synchronization

Overview of Product Data Synchronization

Oracle Product Data Synchronization for Global Data Synchronization Network (GDSN) and UCCnet Services enables companies to securely deliver product information to UCCnet and via the Global Data Synchronization Network (GDSN), and then to syndicate that data to trading partners.

Product Data Synchronization (PDS) comes with pre-built item attributes that map to the attribution defined for product data synchronization by GDSN and UCCnet. PDS also provides extensive validation checks against the approved formats for these attribute definitions to help assure that only clean data is sent. Besides, PDS provides a complete messaging solution that is built to the format mandated by GDSN and UCCnet.

PDS is designed to help make sure that companies avoid both the direct and indirect costs of sending incorrect product data to UCCnet and their trading partners. Direct costs include the fines imposed by UCCnet for sending 'dirty data', and indirect costs include delayed or lost sales, etc. PDS' item attributes come with extensive data validation checks to make sure that input data is clean. The messaging system provides a complete transaction history of all messages to provide a full audit trail of all communications with data pools and trading partners.

Registration and Syndication
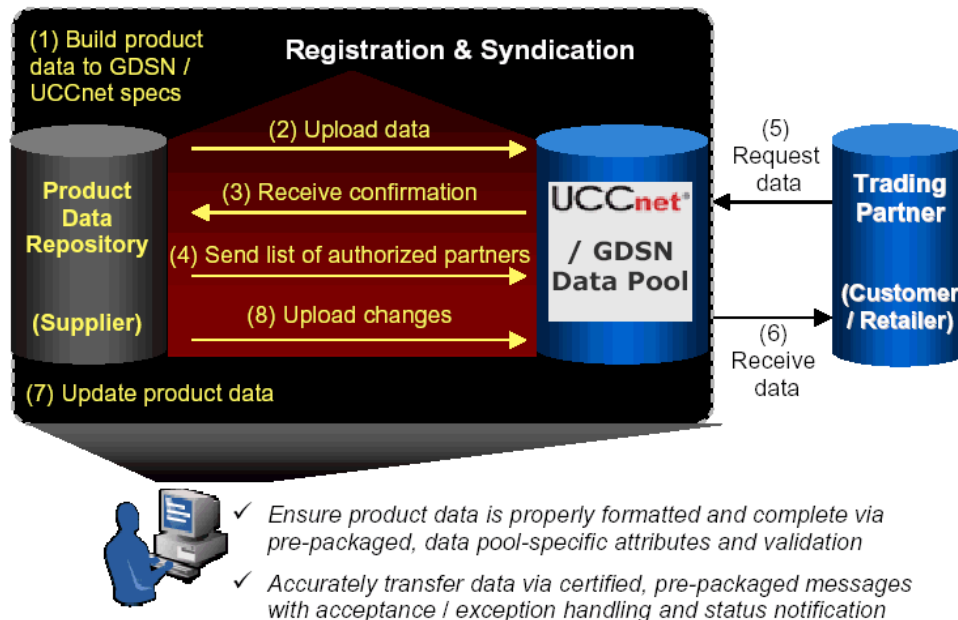
Overview of Global Data Synchronization Network and UCCnet

The Global Data Synchronization Network and its Global Registry provide a foundation for electronic commerce by providing Internet-based product information registration and synchronization services. These services include compliance verification, synchronization of product information, and registry and life cycle management of synchronized products, user locations, and user trade capabilities. These services allow companies to synchronize item information across their entire supply chain. Manufacturers can register and update their product information with GDSN through standard processes, and any of the manufacturer's trading partners – distributors, sellers, retailers, etc. can access that information.

Oracle PDS allows companies to efficiently synchronize accurate product data with customers and suppliers. PDS provides companies with a complete solution for managing their product information and then for delivering their relevant product data to their trading partners by enabling them to register Global Trade Item Numbers (GTINs) associated with one or many customer sites (Global Location Numbers) to UCCnet.

The key features of PDS are:

- Author and manage items/GTINs, attributes and associated packaging hierarchies

- Manage registration process of items/GTINs to the Global Registry (GDSN/UCCnet)

- Manage trading partner subscriptions

- Manage publication/syndication of items/GTINs to trading partners

- Manage ongoing changes to product data and appropriate re-syndications to trading partners

Global Data Synchronization Network

Profile Options

Oracle Advanced Product Catalog (APC) provides several system profile options to enable or change the behavior of certain features in the application.

Following is a description of the GDSN profile option:

APC Profile	Description	Default Value
EGO GDSN ENABLED	Governs the GDSN Synchronization functionality. Changing the value to Yes enables GDSN functionality. Once you turn the flag to yes, then: • GDSN default functional area appears • GDSN attributes appear in setup list page • GDSN attributes get associated to Item Catalog Category • GDSN Syndicated functional area flag on the Create Item and Functional Classification page gets enabled.	Null

Lookups

Oracle APC provides system level lookups to store values for the entire site.

APC Look up	Description	Default Value
EGO_UCCNET_DATA_SOURCE_GLN • GLN • Name	This lookup allows you to enter the Information provider GLN number and name. **Important:** To specify the GLN and the name of the Information Provider, you need to update a lookup: LOOKUP_TYPE = EGO_UCCNET_DATA_SOURCE_GLN LOOKUP_CODE = GLN Meaning = *Actual GLN* LOOKUP_CODE = NAME Meaning = *Actual Name*	Null

Setting up GDSN Catalog

GDSN Catalog is similar to other functional area default catalogs that exists in the Oracle E-Business for example Inventory and purchasing catalog. When you associate an Item/GTIN to GDSN Syndicated functional area, Item/GTIN gets assigned to default Catalog associated to the GDSN Syndicated.

You can associate an existing catalog to this functional area or you can create a new Catalog and associate it.

UCCnet data pool requires use of UDEX Catalog as its Item/GTIN Classification. For example, to communicate with UCCnet data pool, you can create the GDSN catalog using the UDEX catalog category taxonomy. UDEX Category follows a 3 segment flex structure.

The entire UDEX catalog can be set-up in the GDSN catalog or just the categories that are needed as per your business requirements. For example, 'Safety/Security & Surveillance/Survival' has two sub categories: 'Security Other', and 'Fire / Chemical Safety products'. If the company makes Lock and does not make any fire or chemical safety products then the catalog Administrator can setup only one child category as 'Security Other'.

For details on creating catalogs, see the Defining Catalogs section.

You can then use the Import from excel Item Catalog Category assignments functionality to mass assign Item/GTIN to this GDSN Catalog.

GDSN Syndication Functional Area

Functional Area	Catalogs	Description
Inventory	Inv.Items	Inventory Category Set
Purchasing	Purchasing	Purchasing Category Set
Planning	Inv.Items	Inventory Category Set
Service	Inv.Items	Inventory Category Set
Costs	Inv.Items	Inventory Category Set
Engineering	Inv.Items	Inventory Category Set
Order Management	Inv.Items	Inventory Category Set
Product Line Accounting	Inventory	Inventory Category Set
Asset Management	Enterprise Asset Management	Enterprise Asset Management Category Set
Contracts	Contracts	Category Set for Contract Item Categories
Product Reporting	Product	Product Category
GDSN Syndication	GDSN Catalog	GDSN Catalog

Setting up GDSN Attributes

Oracle Product Data Synchronization seeds 150+ GDSN attributes in various GDSN Attribute Groups. GDSN attribute groups are similar to the existing User-defined attribute group functionality.

List of seeded GDSN Attributes

Select All | Select None

Select	Display Name	Internal Name	Description	Attribute Group Type	Multi-Row	Database View	Edit
☐	Bar Code	Bar_Code	UCCnet: Bar Code	Item GTIN Multi Row Attributes	Yes		✎
☐	Country Of Origin	Country_Of_Origin	UCCnet: Country Of Origin	Item GTIN Multi Row Attributes	Yes		✎
☐	Date Information	Date_Information	UCCnet: Date Information	Item GTIN Single Row Attributes	No		✎
☐	Delivery Method Indicator	Delivery_Method_Indicator	UCCnet: Delivery Method Indicator	Item GTIN Multi Row Attributes	Yes		✎
☐	FMCG: Identification	FMCG_Identification	UCCnet: FMCG Identification	Item GTIN Single Row Attributes	No		✎
☐	FMCG: Marking	FMCG_MARKING	UCCnet: FMCG MARKING	Item GTIN Single Row Attributes	No		✎
☐	FMCG: Measurements	FMCG_Measurements	UCCnet: FMCG Measurements	Item GTIN Single Row Attributes	No		✎
☐	GTIN Color Description	Gtin_Color_Description	UCCnet: GTIN Color Description	Item GTIN Multi Row Attributes	Yes		✎
☐	Handling Information	Handling_Information	UCCnet: Handling Information	Item GTIN Single Row Attributes	No		✎
☐	Handling Instructions	Handling_Information	UCCnet: Handling Instructions	Item GTIN Multi Row Attributes	Yes		✎
☐	Hazardous Information	Hazardous_Information	UCCnet: Hazardous Information	Item GTIN Multi Row Attributes	Yes		✎
☐	Manufacturing Information	Manufacturing_Info	UCCnet: Manufacturing Information	Item GTIN Multi Row Attributes	Yes		✎
☐	Material Safety Data	Material_Safety_Data	UCCnet: Material Safety Data	Item GTIN Single Row Attributes	No		✎
☐	Order Information	Order_Information	UCCnet: Order Information	Item GTIN Single Row Attributes	No		✎
☐	Packaging Marking	Packaging_Marking	UCCnet: Packaging Marking	Item GTIN Single Row Attributes	No		✎
☐	Price Date Information	Price_Date_Information	UCCnet: Price Date Information	Item GTIN Single Row Attributes	No		✎
☐	Price On Trade Item	Price_Information	UCCnet: Price On Trade Item	Item GTIN Single Row Attributes	No		✎
☐	Size Description	Size_Description	UCCnet: Size Description	Item GTIN Multi Row Attributes	Yes		✎
☐	Size Description	Uccnet_Size_Description	UCCnet: Size Description for the Item	Item GTIN Single Row Attributes	No		✎
☐	Temperature Information	Temperature_Information	UCCnet: Temperature Information	Item GTIN Single Row Attributes	No		✎
☐	Trade Item Description	Trade_Item_Description	UCCnet: Trade Item Description	Item GTIN Single Row Attributes	No		✎
☐	Trade Item Harmonized System Identification	TRADE_ITEM_HARMN_SYS_IDENT	UCCnet: Trade Item Harmonized System Identification	Item GTIN Multi Row Attributes	Yes		✎
☐	Trade Item Hierarchy	Trade_Item_Hierarchy	UCCnet: Trade Item Hierarchy	Item GTIN Single Row Attributes	No		✎
☐	Trade Item Marking	Trade_Item_Marking	UCCnet: Trade Item Marking	Item GTIN Single Row Attributes	No		✎
☐	Trade Item Measurements	Trade_Item_Measurements	UCCnet: Trade Item Measurements	Item GTIN Single Row Attributes	No		✎

Select Object: [Generate Database View] [Delete]   ◎ Previous 1-25 ▾ Next 25 ◉

You can customize these GDSN Attributes within the GDSN attribute groups. For example –

- You can create value sets and associate it to the attribute as appropriate. For example if you have few selected brand names, you can create a value set for the brand names and associated to the Brand Name attribute.

- Certain Value sets are seeded. For example EANUCC type. If you only use certain types you can customize the value set by disabling the not required ones.

- You can specify default values for any attribute.

- You can disable these GDSN attributes. For example, Drained Weight is an attribute in the Trade Item Measurements attribute group. Drained weight is not a mandatory GDSN attribute and may not be needed for a company in the Hardlines business.

Customer GDSN Attribute

Following are some of the GDSN attribute setups you might want to evaluate per your business nee:

Attribute Group	Attribute Name	Unit of Measure Class
Trade Item Measurements	Diameter	Length
Trade Item Measurements	Gross Weight	Weight
Trade Item Measurements	Drained Weight	Weight
Trade Item Measurements	Peg Horizontal	Length
Trade Item Measurements	Peg Vertical	Length
Order Information	Order Lead Time	Time
Handling Information	Maximum Stacking Weight	Weight
Hardlines Industry	Pieces Per Trade Item	CNT
Hardlines Industry	Nesting Increment	CNT

Setup of Value sets for the following attributes:

Attribute Group	Attribute
Trade Item Description	Coupon Family Code
Trade Item Measurements	Net Content Value – Net content Unit of Measure
GTIN Color Description	Color Code Description
GTIN Color Description	Color Code List Agency
Size Description	Size Code List Agency
Size Description	Size Code Value
Handling Information	Handling Instructions Code
Hazardous Information	Class Of Dangerous Code
Hazardous Information	Dangerous Goods Hazardous Code
Hazardous Information	Dangerous Goods Regulation Code
Hazardous Information	Dangerous Goods Packaging Group
Trade Item Harmonized System Identification	Harmonized Tariff System Identification Code

PDS supports hardlines and FMCG industry extensions. These industry extension attributes are not associated to the Item Catalog Category when the EGO GDSN ENABLED profile option is changed to yes. You need to associate these attributes to the Item Catalog category and then create or update the existing page to display the hardlines or the FMCG attributes.

Setting Up Packaging Structure

Oracle APC seeds Packaging BOM as the structure type. Packaging configuration that is created for this structure type is being sent to GDSN data pool as part of the publication message. You can create Structure name for the Packaging Hierarchy for this structure type.

User can use this structure type and structure name to create their packaging configurations.

For more information about structure type and structure names, see the Defining Item Structures section.

Setting up GTIN Number Generation

The Global Trade Item Number (GTIN) encompasses existing UPC and EAN identifiers, uniquely identifying all levels of the product hierarchy and delivering data in a consistent format and structure. The GTIN is a 14-digit number that contains information about your company and item, as well as how it is packaged.

Oracle Product Data Synchronization provides the ability to automatically generate and assign GTIN to your product depending on the user-defined function or allow users to enter by providing free form number field.

Depending upon your User-defined functions, the product classification or attributes like UPC or EAN codes can be set as function input parameters.

APC validates the GTIN check digit, during GTIN creation process.

GTIN Number Generation

Setting up GLN for Customers

GLN is the Global Location Number that is assigned to a company for any of its physical locations. GLN is 13-digit number with last digit being a check digit. Oracle allows you to assign GLN numbers to each customer addresses using Customer Online screens.

Setting Up GLN

Global Location Number is not displayed by default, as it is primarily used for the purpose of Product Data Synchronization. Using personalization framework you can display the field.

> **Note:** If the Global Location Number field doesn't appear on the update customer address screen, then please use OA personalization option to display it.

To display Global Location Number:

1. Using system administrator responsibility set the profile option "Personalize Self-Service Defn" to Yes.

2. Login using your user id and navigate to profile option and set the "FND: Personalization Region Link Enabled" to Yes.

3. Navigate to the Customer Update Address screen page for any customer using Customer Online Super User responsibility.

4. Click the personalize region link next to the country list of values attribute.

5. On the Choose Personalization Context page accept the default values for the fields and click apply.

6. In the Personalization Structure section, click the personalize icon for "RowLayout: (glnRowLayout)"

7. In the Personalize properties section, for Rendered row, change the site setting to true. (Notice: Original Definition is set to False).

8. Click Apply to save the changes

Managing Global Trade Item Number Information

> **Important:** To use these features except GTIN Cross References, you must have licensed Oracle Product Data Synchronization for GDSN and UCCnet Services. You must also set the "EGO GDSN Enabled" profile option to "Yes."

Any item in your item catalog can be enabled for GDSN syndication by assigning it to the catalog associated with the GDSN Syndication Functional Area. Once an Item is classified as GDSN Syndicated, GDSN attributes are displayed and you can begin defining GDSN attributes.

To create an item for GDSN syndication:

1. On the create Item page, select Item Catalog Category or existing Item.

2. Select GDSN syndicated

3. On the Create Item: Enter Item/Description page, select Trade Item Unit Descriptor.

4. Add or copy GDSN attributes in the copy step.

5. In the Review Item information, view and update GDSN attributes.

6. In the Categories sub tab, view default GDSN Catalog and category assignments. You can update the category assignment, but cannot delete the GDSN Catalog assignment.

7. In the Association sub tab, add GTIN Cross-References. To find details on how to add GTIN cross-references, see "Managing GTIN Cross-References" in the Oracle Advanced Product Catalog User's Guide.

   > **Note:** The GDSN Syndicated flag applies only when the Item you are creating is an approved Item and doesn't require a New Item Request approval.
   >
   > The GDSN Syndicated flag does not apply when creating multiple Items.

To enable an existing Item for GDSN syndication:

1. Find the item that you want to enable for GDSN syndication.

2. On the Item's overview page, click the Classification and Categories tab.

3. Check whether the item has been already enabled.

4. Click Update

5. On selecting the GDSN Syndicated check box, the categories assignments region gets refreshed with GDSN Catalog and default category assignment.

6. You can update the category assignment, however you cannot delete the GDSN Catalog assignment.

7. Adding a GDSN Catalog Category assignment automatically selects the GDSN Syndicated checkbox.

8. Click Apply to save your changes.

> **Note:** Once you enable an Item for GDSN Syndication, you cannot remove it. You cannot de-select the GDSN Syndication, nor delete the GDSN Catalog Category assignment. You can update the GDSN Catalog category assignments to a new GDSN category.
>
> Import of Item Catalog Category assignments to the GDSN Catalog using Microsoft Excel automatically enables Items for GDSN Syndication.

Oracle UCCnet Registry Messages

Global Registry is a repository for registry messages. Oracle Messages for UCCnet Registry offers a bundle of pre-configured, certified UCCnet XML messages to support Supplier and Retailer data synchronization, along with message management tools to give customers a 360 degree overview of the electronic exchange. These messages allow subscribers to register item attribute information as well as send and receive requests and notifications. XML Gateway, in conjunction with the Workflow business event system (WF BES) is used to generate the messages based on the user initiated actions. Specific XML Gateway maps are built for each of the UCCnet specified XML messages which are then used to communicate with UCCnet registry. UCCnet also provides XML schema definitions to allow validation of the XML messages , prior to their propagation to UCCnet for the necessary action.

Oracle Messages For UCCnet Registry is tightly integrated with Oracle Advance Product Catalog (APC). APC is the base application that stores item information by GTIN, and GTIN hierarchy that mimics UCCnet data structure, for manufacturing and retail companies. APC is the initiation point for the Registry messages if the customers are Suppliers.

> **Note:** The first release of Oracle Messages for UCCnet Registry supports the Supplier Synchronization Suite. Support for the Demand Synchronization Suite will be introduced in the next major release of this product.

> **Important:** For details on implementing Oracle UCCnet Registry Messages, see *Oracle Applications Installation Guide.*

> **Important:** For details on installing the Messages for UCCnet (M4U) patch, see Oracle*MetaLink* note number 298620.1

The following pre-built XML messages are used in the supply-side functionality in UCCnet

- Register Catalog Item Registry (RCIR)
- Catalog Item Notification (CIN)
- Query Worklist Messages

Register Catalog Item Registry

Register Catalog Item Registry (RCIR) messages add products to the UCCnet data pool. It updates the brand owner registry if it is the first occurrence. Possible actions are add, change, correct, and delete. Using RCIR, items can be registered in synchronous or asynchronous modes. When a single message is sent to the registry, the UCCnet registry response is in the synchronous mode. However, when multiple messages (batch mode) are sent, only an acknowledgement is received from the UCCnet registry synchronously. To obtain processing details of the batch, the query message transaction is initiated.

Catalog Item Notification

Catalog Item Notification (CIN) messages are delivered to the recipient. Since each CIN is targeted to a single recipient, CIN allows individual values of the product. It includes extensions as well as industry specific attributes like those for hardlines, FMCG etc. Possible actions for CIN are add, change, correct, delete, and withdraw.

Query Worklist Messages

Query messages are used to retrieve information from the GLOBALRegistry or notifications received from trading partners/recipients. XML schema messages are used to transmit the query request. The different query messages used are RCIR in batch mode, Request For Catalog Item Notification (RFCIN), and CIC. Besides, GTIN query is also used.

RCIR in batch mode is used to collect the processing status for asynchronous messages back from registry.

RFCIN is an XML message request for information pertaining to an item. A UCCnet recipient trading partner (commonly a retailer) who wants to examine the attributes of an item that has been published in the GDSN (Global Data Synchronization Network), but does not yet want to set up a permanent subscription, sends an RFCIN to request a one-time publication of item information from a source trading partner (commonly a supplier) within the GDSN. The GTINs (Global Trade Item Numbers) included in the RFCIN would have been registered in the UCCnet Data Pool (called 'GS1 USA') and the GS1 Global Registry. The recipient trading partner sends an RFCIN to the UCCnet Data Pool. Item data can be requested for one or more of the following criteria:

- GTIN
- Category
- Target Market
- Global Location Number (GLN)

The UCCnet Data Pool receives the RFCIN information, and forwards a copy to the source trading partner. In case the supply trading partner uses a different data pool, UCCnet Data Pool submits the RFCIN message to the GS1 Global Registry. In response to an RFCIN, the source trading partner submits a Catalogue Item Notification (CIN) to UCCnet. CIN contains Item attribute details, including those of industry

extensions. UCCnet receives the CIN document and sends it to the recipient trading partner's Worklist. For the first thirty-day period after submission of the RFCIN message, the recipient trading partner will receive item information including changes and/or corrections to the items. If the recipient submits a "Reject" confirmation action prior to the expiration of the thirty-day period, changes to the item data will no longer be received. If no action is taken within the thirty-day period, the recipient trading partner will continue to receive the requested Catalogue Item data, even after the thirty-day period has ended, unless a Catalogue Item Confirmation (CIC) message with an action of "Reject" is submitted in response to the CIN. It is the recommended practice for recipient trading partners to submit a CIC message as soon as they decide on the confirmation action they want to take viz. Accepted, Synchronized, Rejected, or Review. The GS1 Global Registry validates the message, determines the Source Data Pool to route the message to, and delivers the message appropriately.

GTIN Query retrieves the elements defining the GTIN from the UCCnet registry.

Support UCCnet Supplier Synchronization Suite

The initial release of Oracle Messages for UCCnet Registry offers the entire Supplier Synchronization Suite. Oracle customers who are suppliers or manufacturers use the entire Synchronization Suite to send item information to UCCnet or to retrieve synchronization feedback from their trading partners. Oracle Messages for UCCnet Registry is certified on 2.3 version of Supplier Synchronization Suite.

The following figure shows the components of the Oracle solution for UCCnet integration for suppliers. Advanced Product Catalog is the base application where item information is stored. Oracle Messages for UCCnet Registry and Supply Chain Trading Connector are the messaging components of the integration. 10g PartnerConnect provides the AS2 protocol to transport Registry messages to UCCnet and to receive Registry messages from UCCnet.
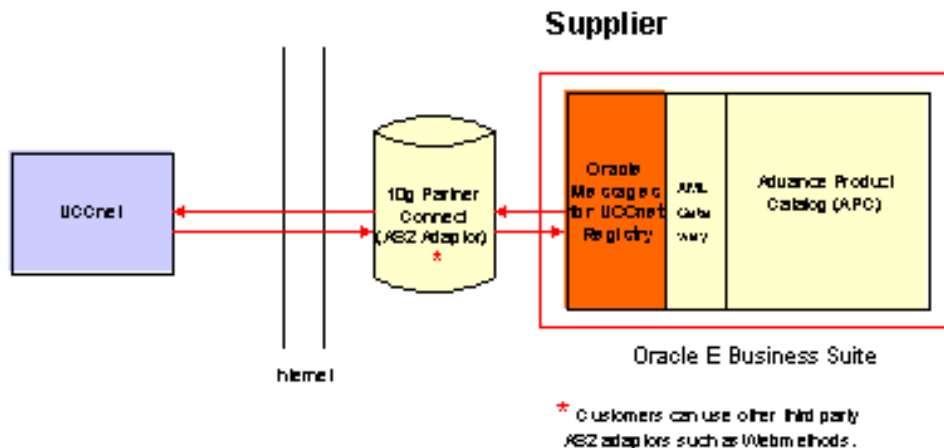
The following provides further details on the UCCnet Supplier Synchronization Suite:

• Register Catalogue Item Registry (RCIR) message is the message for registering an item in UCCnet Registry or updating core attributes of a registered item. Add, Change and Modify features are supported for this message.

- Catalogue Information Notification (CIN) message is a notification from manufacturers/suppliers to their Trading Partners to notify them of new item registration (GTIN) or item information change.

- Catalogue Item Confirmation (CIC) message is the message returned by the Trading Partner in response to a CIN.

- Request for Catalogue Information Notification (RFCIN) is the message sent by a trading partner to a Supplier to request a CIN for specific GTINs.

- Direct Store Delivery

- Market Group Linking

- Hardline Industry Extension support, other industry extensions are in consideration for future release.

Supply Chain Trading Connector

In addition to the registry messages, Oracle Messages for UCCnet Registry uses a monitoring/auditing tool called Supply Chain Trading Connector. This tool sits outside of XML Gateway, monitors and audits all incoming and outgoing UCCnet messages for Oracle E-Business Suite. Two features of the tool Collaboration History and Notification Processing are used by the Oracle Messages for UCCNET Registry product. Each message exchange is defined as a collaboration in this tool, and is modeled by pre-seeded events and shown as detailed steps in the Collaboration History. When an exception occurs due to message failure or data validation, exception reports are initiated by Notification Processing to notify the Business Users of errors.

The Supply Chain Trading Connector is integrated in a seamless manner with the Advanced Product Catalog to provide Business Users a comprehensive view of the UCCnet information and the exchange of the messages. From the APC item page, a Business User can view the Collaboration status or jump to the Collaboration History to view the detailed information of that Collaboration. From the Collaboration History screen, a Business User can view the item history in APC after viewing the detailed Collaboration information by jumping to the APC item information page.

> **Note:** For details on Supply Chain Trading Connector, see the *Oracle Supply Chain Trading Connector Implementation Guide.*

Collaboration History

Due to the nature of B2B standards, it is often difficult for business end users to understand the status of their trading activity. They often approve something that they know is sent to a trading partner but do not know the status. It is a complicated issue due to the number of messages and acknowledgements sometimes exchanged, as well as, the number of software components that processed the messages for a specific collaboration.

The Collaboration History addresses this challenge by tracking all the events and messages exchanged during a collaboration and making it available to the business end user for inquiry. Using Collaboration History you can track all the messages exchanged, any detailed information including exceptions, and the XML Payloads where available.

Major Features of Collaboration History

- Records all key processing events of a unique collaboration or PIP from the initial event or message, receipt or send, to the final message including any exceptions issued by the trading partner or the B2B Gateway in use.

- Enables users to search history by business document number, date, status, document owner, or trading partner

- Permits you to re initiate a collaboration or PIP if allowed by the collaboration

- Enables users to view the XML Payloads associated with a particular collaboration

- Provides Workflow events to create, update, and add details collaboration history for use when implementing a collaboration

    Note: For further details on Collaboration History, see *Oracle Supply Chain Trading Connector Implementation Guide*

Notification Processing

The major features of Notification Processing are:

- To capture notification messages from sources outside the applications such as a B2B Gateway or Trading partner.

- To capture Exceptions generated by the XML Gateways

- To record notifications in Collaboration History

- To perform actions based on the Notification settings. Valid actions are:

    - Call API

    - Notify Document Owner

    - Notify Trading Partner

    - Notify Administrator

    - Raise Business Event

    - Start Workflow

- To provide predefined Notification Actions for all supported Collaborations.

- To permit configuration of the predefined Notifications to fit your business requirements.

- The ability to establish Notification Actions for new Collaborations defined in Collaboration History.

Notification Action Definition
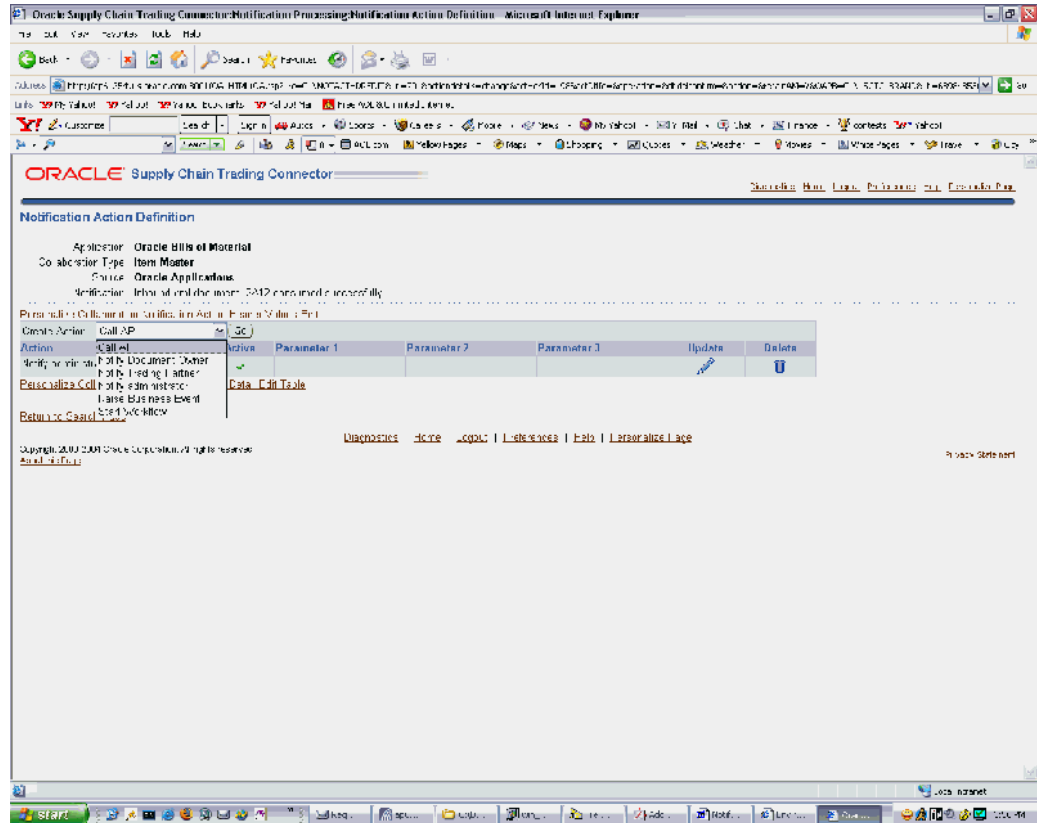
Notification Processing is organized into Notification Types, Notifications and Notification Actions. Notifications are raised to represent the success, exception or error at predefined points within a Collaboration. A Notification can be classified as one of 3 types based upon the Notification Source that raised it: Application, XML Gateway or B2B Gateway. A Notification Action defines the action to be taken at a predefined point within a Collaboration based upon the Notification raised.

Notification Actions

A Notification Action has the following characteristics:

- A Collaboration is an occasion where messages are communicated between an enterprise using the Oracle E-Business Suite and a trading partner usually using the internet, or, between an E-Business Suite application and another application. Collaborations represent business processes such as an Order, Change Order, Forecast, or Work in Process action.

- An Application that owns the collaboration. For example Purchasing or Order Management.

- A Notification Source, which describes the software component generating the Notification.

- A Notification to describe the success, error, or warning.

  Important: All Notification Actions required for Supply Chain Trading Connector supported Collaborations have been pre-configured. All pre-configured Notification Actions that handle error conditions are

pre-configured with the default action of notifying the Administrator. It is expected that additional business rules are specified, using the Notification Processing system, such as initiating a workflow to place an inbound Purchase Order on hold when, for example, a Trading Partner Identification Error is encountered in a given Collaboration

To create new notifications:

After first determining the messages you plan to receive in the new collaboration, you need to define any new Notifications that are required and finally you define the Notification Actions to be performed when these Notifications are raised for the new Collaboration.

1. Navigate to the **Notification Definition** window.

2. Click **Create Definition**.

3. Choose a Notification Source from the drop-down list.

4. Enter the Notification. This is programmatic value or code that you system passes to representing the notification.

5. Enter Description. This is a textual description of what the Notification means.

6. Click **Apply.**.

> **Note:** For further details on Notification Processing, see *Oracle Supply Chain Trading Connector Implementation Guide*

Publish the Registered Information to Trading Partners

Users can register items in UCCnet and publish the item to trading partners. Using Product Data Synchronization for GDSN and UCCnet Services, you can publish Item-GTIN information to customers to inform them of:

- New Item-GTINs being offered

- Changes to existing Item-GTINs

- Withdrawal of Item-GTINs from a specific customer, or delisting of Item-GTINs from the market.

Product Data Synchronization for GDSN and UCCnet Services is an efficient platform for all Item-GTIN publication processing and messaging, allowing you to:

- Find all new or updated Item-GTINs that need to be published to customers

- Track publication event history, with transaction disposition, for each Item-GTIN customer combination

Product Data Synchronization for GDSN and UCCnet Services allows you to publish a single Item-GTIN, or multiple Item-GTINs in a batch. Upon 'Publication', a CIN message is sent to your customer for each Item-GTIN via a GDSN data pool. Your customer can review each message, and decide whether to 'Accept', 'Reject', 'Synchronize' or 'Review' the Item-GTIN. The customer's response is returned via a confirming CIC message, and is stored in your work list on the GDSN data pool. These global registry messages are transferred from the data pool to Product Data Synchronization for GDSN and UCCnet Services, where the message content and disposition status can be viewed for each Item-GTIN customer combination. Disposition statuses are:

- Processed

- Failed

- Customer-Accepted

- Customer-Rejected

- Customer-Review

- Customer-Synchronized

> **Note:** You will need Publication Management function security, View Item privilege, and View Customer Item cross-reference privilege in the Master Organization to perform the following actions on an Item-GTIN for a customer:
>
> - Initial load
>
> - Publish
>
> - Withdraw

You must set up customer subscriptions for each Item-GTIN before you begin the publication process. The publication status for a customer indicates whether the Item-GTIN requires publication. The publication validation status on the GDSN Attribute pages and syndication page indicates whether the Item-GTIN packaging hierarchy is compliant for publication.

> **Note:** Always publish from the top Item-GTIN of a valid packaging hierarchy to a customer, as the entire packaging hierarchy must be included in the CIN message.

A valid packaging hierarchy is one that contains at least one consumable and one orderable GTIN. A Top GTIN is an item that is not a component in a packaging hierarchy, that has at least one orderable and one consumable unit in it's packaging hierarchy.

To publish an Item-GTIN to customers:

1. From the top Item-GTIN page, navigate to the Customer Syndication page. This page displays the list of customers to whom you have already published or plan to publish the Item-GTIN packaging hierarchy.

   > **Note:** For details about creating and maintaining customer syndication, see "Managing Customer Syndication".

2. For each customer, there is a publication status and a corresponding disposition status. From the Publication status, you can determine which customers the Item-GTIN has not been published to, or requires re-publication. Publication statuses are:

   - Published

   - Not Published

   - Re-publication Needed

   - Publication In Progress

- Withdrawn

- Delisted

- Rejected

> **Note:** You cannot publish information to customers to whom you have previously published that information if no new GDSN attribute changes have occurred since your last publication.

3. Click the Publish action icon to publish the Item-GTIN to a customer. On the Publish Items page, you can view the entire packaging hierarchy and the list of customers for each packaging hierarchy.

4. You can review the name and GLN of the data pool to which the Item is being published, as well as the customer's name, address and GLN.

5. After you submit the Item-GTIN for publication, a CIN message is sent for each customer that the Item-GTIN is being published to. A message group id is generated to track all Item-GTIN publication messages sent in this transaction.

6. The Disposition status for each message is updated with the response that is received from the customer via the data pool.

> **Note:** If a customer rejects an Item-GTIN publication, Product Data Synchronization for GDSN and UCCnet Services will not allow you to re-publish the Item-GTIN to the same customer.

> **Note:** A first time publication message to a customer is sent with a 'New Item' or an 'Initial Load' action. Subsequent re-publication messages are sent with a 'Data Change' action. Product Data Synchronization for GDSN and UCCnet Services automatically derives these actions.

> **Note:** When you update the Item-GDSN attributes for a published Item, its publication status is automatically changed to 'Re-publication Needed' for all customers. However, if you update a customer-specific Item-GDSN attribute, then the publication status is automatically changed to 'Re-publication Needed' for that particular customer only.

To withdraw an Item-GTIN from a customer:

The withdraw action allows you to temporarily discontinue an Item-GTIN synchronization process with one or more customers. Once you have successfully withdrawn the top Item-GTIN from a customer, the Item-GTIN is not published to the customer until you explicitly re-publish the Item-GTIN to this customer.

> **Note:** You can only withdraw Item-GTINS whose publication status is 'Published' and disposition status is 'Not Rejected'.

1. From the top Item-GTIN page, navigate to the Customer Syndication page. This page displays the list of customers to whom you have already published or plan to publish the Item-GTIN packaging hierarchy.

2. For the customers that you have already published to, select those from you which you want to withdraw the Item-GTIN, and click Withdraw.

3. The Withdraw from Customers page allows you to view the entire Item-GTIN packaging hierarchy and the customers from whom you are withdrawing the Item-GTIN.

4. A CIN message is sent to each customer from whom the Item-GTIN is being withdrawn. A message group id is generated to track all messages sent in this transaction

To delist an Item-GTIN from customers:

The delist action permanently discontinues the synchronization process with all customers for this Item-GTIN. Once you successfully delist the top Item-GTIN, Product Data Synchronization for GDSN and UCCnet Services does not allow you to publish this Item-GTIN to any customer.

> **Note:** You can only delist the Item-GTINS whose publication status is published and disposition status is not rejected.

1. From the top Item-GTIN page, navigate to the Customer Syndication page. This page displays the list of customers to whom you have already published or plan to publish the Item-GTIN packaging hierarchy.

2. Click Delist from the customer syndication page.

3. All customers to whom the Item-GTIN was previously published will be delisted.

4. A CIN message is sent to each customer from whom the Item-GTIN is being delisted. A message group id is generated to track all messages sent in this transaction.

To publish multiple Item-GTINS to customers:

The publication workbench allows you to view multiple Item-GTINS for a customer. Similar to the registration workbench, you can search for Item-GTINS that are associated with the customer.

> **Note:** You can only search for top Item-GTINS in the publication workbench.

You can Publish, Withdraw, and Delist Item-GTINS from this workbench. These actions are described in the section To publish an Item-GTIN to customers, .

1. On the Publication Management page, select the Item-GTINS you want to publish and click Publish.

2. On the Publish to Customers page, a list of all the top Item-GTINS you have selected is displayed with its entire packaging hierarchy.

3. Review the list of customers per each Item-GTIN packaging hierarchy.

    You can also select Item-GTINS for Withdraw and Delist.

    On the publication workbench, you can select top Item-GTINs and click Check Data Validation to see whether the Item-GTIN packaging hierarchy you plan to publish is compliant.

To publish an Item-GTIN to customers with Initial load action:

If a supplier who has recently joined a data pool, such as UCCnet, wants to establish item synchronization with an existing Demand Side customer through the data pool, they can use the Initial Item load publication to synchronize data for items currently being traded between the two organizations.

1. On the Publication Management page, find the Items you want to publish.

2. Select the Items you want to publish with the Initial load action, click Initial load.

3. On the Publish (Initial Load) to Customers page, a list of all the top Item-GTINS you selected is displayed with their packaging hierarchy.

4. Review the list of customers for each Item-GTIN packaging hierarchy.

> **Note:** Initial load action is only performed for the first time publication with the customer.

A

Change Management Workflows

This appendix covers the following topics:

- Change Management Workflows

Change Management Workflows

Because Oracle Advanced Product Catalog (APC) uses Oracle Workflow technology to automate business processes in Change Management and Group Member registration (among other tasks), you must set up Oracle Workflow to implement Oracle Advanced Product Catalog (APC).

Workflow Versions

Oracle Product Lifecycle Management Family Pack C includes FND patch 3262159 (11i.FND.H mini-pack) which includes 11i.OWF.H minipack (3258819)- as a prerequisite.

> **Note:** Run wfver.sql under $FND_TOP/sql for detailed information about workflow versions.

Workflow Setup

Refer the Metalink Note: 258312.1 "About Oracle Workflow Mini-pack 11i.OWF.H" This document lists the prerequisite patches and configuration information. It also lists Oracle Workflow Implementation and User Documentation required for setup. You need to complete Oracle Workflow Setup before using Oracle Product Lifecycle Management Family Pack C Workflow functionality.

Overview of Change Management Workflows

Change Management uses Oracle Workflow technology to execute workflow routing for a change object and events occurring in a change object (such as Submit, Reassign, Priority Change, and Status Change actions). Change Management enables you to model your business processes in terms of generic order processes. So you can model your business processes by customizing or extending seeded workflow processes and/or modifying custom hook PL/SQL procedures.

Viewing Processes in Oracle Workflow Builder

From within the Oracle Workflow Builder you can view the different workflow processes and their associated functions, messages, subprocesses, notifications, and properties.

To view processes in Oracle Workflow Builder:

1. Within Oracle Workflow Builder, select Open from the File menu and connect to the database. Alternatively, you can connect to the workflow definitions file engchgmt.wft located in the product directory tree of your Oracle Applications server.

2. Expand the data source, and then select the item type branch within that data source.

3. Expand the processes branch within your item type and select a process activity to display the diagram of the process in a Process window.

Display Name (Internal Name)	Purpose	Location of Flat File Store on Server (Location of other Flat File Stores with Item Type embedded)
ENG: Standard(ENGWFSTD)	Contains a collection of commonly used (by other Eng Workflow Item Types) Function Activities and Lookup Types.If the activity will be shared by more than one object's process, create the activity in the ENG: Standard (ENGWFSTD) item type. Otherwise, create the activity in the appropriate object's item type	$ENG_TOP/patch/115/import// engchmgt.wft
Change (ENGCACT)	Contains activities, attributes, notifications, and processes used by Change Action Process	$ENG_TOP/patch/115/import// engchmgt.wft
Change Line (ENGCLACT)	Contains activities, attributes, notifications, and processes used by Change Line Action Process	$ENG_TOP/patch/115/import// engchmgt.wft
Change Workflow Routing(ENGCRT)	Contains activities, attributes, notifications, and processes used by Change Approval Routing Process	$ENG_TOP/patch/115/import// engchmgt.wft
Change Workflow Routing Step(ENGCSTEP)	Contains activities, attributes, notifications, and processes used by Change Approval Routing Step Process	$ENG_TOP/patch/115/import// engchmgt.wft

Pre-Seeded Workflow Processes

Change (ENGCACT)

Process Internal Name	Process Display Name	Description
INITIATE_CHANGE	Initiate Change	Sends an assignment notification to assignee and starts "Initiate Change Line" workflow if Change has a line. This workflow is started when users submit a Change Object.
PRIORITY_CHANGE	Priority Change	Sends a priority change notification to Requestor, Owner, and Assignee for changes. This workflow is started when users change the priority for a Change Object.
REASSIGN_CHANGE	Reassign Change	Sends an assignment notification to assignee. This workflow is started when users reassign a Change Object.
REQUEST_COMMENT	Request Comment	Sends a notification to request comment to specified people on the Request Comment Action page. When the recipient responds to the request notification, Workflow sends a response notification to the requestor. This workflow is started when users request comment for a Change Object.
RESPONSE_FYI	Send Response FYI	This process is called when the recipient responds to the request notification in Request Comment workflow process.
STATUS_CHANGE	Status Change	Sends a status change notification to Requestor, Owner, and Assignee for a change. This workflow is started when users change the status for a Change Object.

Change Line (ENGCLACT)

Process Internal Name	Process Display Name	Description
INITIATE_CHANGE	Initiate Change	Sends an assignment notification to assignee. This workflow is started from the Change Header's Initiate Change Workflow or when a user creates a Change Line if Change Header is Open.
REASSIGN_CHANGE	Reassign Change	Sends an assignment notification to the assignee. This workflow is started when a user reassigns a Change Line.
STATUS_CHANGE	Status Change	Sends a status change notification to Header Requestor, Header Owner, Header Assignee, Line Owner and Line Assignee. This workflow is started when a user changes status for a Change Line.

Change Workflow Routing (ENGCRT)

Process Internal Name	Process Display Name	Description
ROUTE_AGENT	Workflow Routing Agent	Controls Workflow Routing and sends Approval Status change notification to Requestor, Owner, and Assignee for a Change. This workflow is started when a user submits an Workflow Routing for a Change Object. Also this workflow is aborted when user aborts an Workflow Routing.

Change Workflow Routing Step (ENGCSTEP)

Process Internal Name	Process Display Name	Description
REQUEST_DEFINITION	Definition	Sends a definition request notification to specified assignees on Workflow Routing page and controls definition request for New Item Request. If the assignees do not respond by the specified period, workflow will also send a reminder notification. Once the step is completed, workflow will return control parent Workflow Routing Agent workflow. This workflow is started from Workflow Routing Agent based on the definition of Workflow Routing.

REQUEST_DEFIINITION_ AND_APPR	Definition and Approval	Sends a definition and approval request notification to specified assignees on Workflow Routing page and controls definition and approval request for New Item Request. If the assignees do not respond by the specified period, workflow will also send a reminder notification. Once the step is completed, workflow will return control parent Workflow Routing Agent workflow. This workflow is started from Workflow Routing Agent based on the definition of Workflow Routing.
REQUEST_APPROVAL	Request Approval	Sends an approval request notification to specified assignees on Workflow Routing page and controls approval request. If the assignees do not respond by the specified period, workflow will also send a reminder notification. Once the step is approved, workflow will return control parent Workflow Routing Agent workflow. This workflow is started from Workflow Routing Agent based on the definition of Workflow Routing.
REQUEST_COMMENT	Request Comment	Sends a comment request notification to specified assignees on Workflow Routing page and controls comment requests. If the assignees do not respond by the specified period, workflow will also send a reminder notification. Once the step is completed, workflow will return control parent Workflow Routing Agent workflow. This workflow is started from Workflow Routing Agent based on the definition of Workflow Routing.
SEND_FYI	FYI	Sends an FYI notification to specified assignees on Workflow Routing page. Once the step is completed, workflow will return control parent Workflow Routing Agent workflow. This workflow is started from Workflow Routing Agent based on the definition of Workflow Routing.

Following are two Process Attributes for the Workflow Routing Step workflow process:

1. **Default Change Role Instance Set (DEFAULT_CHANGE_ROLE)**

   Assigned on current or last Workflow Routing. Implicit grants securely based on the steps workflow process.

   ENG_CHANGE_WF_APPROVERS -> Granted Role: Approver (ENG_CHANGE_ APPROVER)

   ENG_CHANGE_WF_REVIWERS -> Granted Role: Reviewer (ENG_CHANGE_ REVIEWER).

   Implicit Role (Instance Set) to be granted is defined in the Step workflow process attributes

   DEFAULT_CHANGE_ROLE

   In case of seeded Step Workflow Process

   Request Approval (DEFAULT_CHANGE_ROLE: "ENG_CHANGE_WF_AP PROVERS")

   The step assignees get "Approver" implicit role from instance set ENG_CHANGE_WF_APPROVERS.

   Request Comment & FYU (DEFAULT_CHANGE_ROLE: "ENG_CHANGE_WF_ REVIEWERS")

   The step assignees get "Reviewer" implicit role from instance set ENG_CHANGE_WF_REVIEWERS. Oracle Workflow does not support other roles by default.

2. **Activity Condition Code (ACTIVITY_CONDITION_CODE)**

   Attribute will indicate whether or not the Condition radio box is displayed on the Workflow Routing Create/Update step page. Acceptable values are Y or N.

3. **Enable Flag in Type: Definition (DEFINITION)**

   Attribute will indicate whether or not the this process is displayed in the Workflow Process pull-down list based on the Definition Type of Workflow Routing (Template) on the Workflow Routing Create/Update step page. Acceptable values are Y or N.

4. **Enable Flag in Type: Approval (APPROVAL)**

   Attribute will indicate whether or not the this process is displayed in the Workflow Process pull-down list based on the Approval Type of Workflow Routing (Template) on the Workflow Routing Create/Update step page. Acceptable values are Y or N.

5. **Enable Flag in Type: Definition and Approval (DEFINITION_APPROVAL)**

   Attribute will indicate whether or not the this process is displayed in the Workflow Process pull-down list based on the Definition and Approval Type of Workflow Routing (Template) on the Workflow Routing Create/Update step page. Acceptable values are Y or N.

6. **Enable Flag in Type: Generic (GENERIC)**

   Attribute will indicate whether or not the this process is displayed in the Workflow Process pull-down list based on the Generic Type of Workflow Routing (Template) on the Workflow Routing Create/Update step page. Acceptable values are Y or N.

7. **Association Enabled Flag (ASSOC_ENABLED)**

Attribute will indicate whether or not the the association for the assignee is available in the Workflow Routing Step for New Item Request on the Create/Update step page. Acceptable values are Y or N. This atttribute is only applicable for "Definition" and "Definition and Approval" Workflow.

Seeded Workflow Item Attributes

The following table describes the common item attributes for Change Management Workflow. Each Workflow Item Type should have these attributes.

All Item Type

Internal Name	Display Name	Description	Type	Length	Misc info (used for etc)
CHANGE_ID	Change Id	Engineering Change Unique Identifier	Number		Primary Key Values
CHANGE_ NOTICE	Change Number	Engineering Change Number	Text	30	Old Primary Key Values
CHANGE_ NAME	Change Name	Engineering Change Name	Text	240	Token replacement for messages.
ORGANIZAT ION_ID	Organization Id	Organization Id	Number		Old Primary Key Values
ORGANIZAT ION_CODE	Organization Code	Organization Code	Text	3	Token replacement for messages.
CHANGE_ MANAGEMTENT TYPE	Change Management Type	Change Management Type	Text	40	Token replacement for messages.
DESCRIPT ION	ECO Description	ECO Description	Text	2000	Token replacement for messages.
STATUS	Status	Change Status	Text	80	Token replacement for messages.
APPROVAL_ STATUS	Approval Status	ECO Approval Status	Text	80	Token replacement for messages.
PRIORITY	Priority	Priority	Text	50	Token replacement for messages.
REASON	Reason	Reason	Text	50	Token replacement for messages.
ASSIGNEE_ NAME	Assignee Name	Assignee Name	Text	360	Token replacement for messages.

ASSIGNEE_ COMPANY	Assignee Company	Assignee Company	Text	360	Token replacement for messages.
WF_USER_ ROLE	WF Owner User Role	Role of Person Who Started the Workflow	Role		Runtime information for the work item
WF_USER_ID	WF Owner User Id	User Id of Person Who Started the Workflow	NUMBER		Runtime information for the work item
HOST_URL	Host URL and port	Host URL and port	Text		Runtime information for the work item
DEFAULT_ STYLE_ SHEET	Default Style Sheet	Default Style Sheet	Text	100	/OA_ HTML/cabo/ styles/blaf. cssTemporary placeholders for HTML notifications
CHANGE_ DETAIL_ PAGE_URL	Change Detail Page URL	Change Detail Page URL	URL		
MESSAGE_ TEXT_BODY	Message Text Body	Message Text Body	Document		Temporary placeholders for notifications
MESSAGE_ HTML_BODY	Message HTML Body	Message HTML Body	Document		Temporary placeholders for notifications
REVIEWERS_ ROLE	Change Standard Reviewer Role	Change Standard Reviewer Role	Role		Temporary placeholders for notifications
ASSIGNEE_ ROLE	Assignee Role	Assignee Role	Role		Temporary placeholders for notifications
ACTION_ID	Change Action Id	Change Action Id	Number		Primary Key for Change Actions (ENG_ CHANGE_ ACTIONS. ACTION_ID) Runtime information for the work item

ADHOC_ PARTY_LIST	Adhoc Party List	Adhoc Party List	Text		Runtime information for the work item
ADHOC_ PARTY_ROLE	Adhoc Party Role	Adhoc Party Role for Change Action	Role		Temporary placeholders for notifications
RESPONSE_T IMEOUT	Response Timeout	Response Timeout in Minutes for Action calculated from Response_By_ Date	Number		
RESPONSE_ COMMENT	Response Comment	Response Comment	Text		Temporary placeholders for notifications
ATTACH- MENTS	Attachments	Attachments	Document		Temporary placeholders for notifications: Notification Detail Page Attachment LinkFND: entity===
WF_USER_ID	WF Owner User Id	User Id of Person Who Started/ Aborted the Workflow	NUMBER		Runtime information for the work item
WF_USER_ ROLE	WF Owner User Role	Role of Person Who started/ aborted the Workflow	Role		Runtime information for the work item
WF_ADMIN ISTRATOR	WF Administrator	WF Administrator	Role		Constant: SYSADM INFor WFERROR: DEFAULT_ ERROR (Default Error Process.)
WF_SIG_POL ICY	Signature Policy	Signature Policy	Text		To use Signature Policy defaulting in future

ERROR_TIMEOUT	WF Error Timeout Value	Timeout Value in Days	Number		Constant: 0For WFERROR: DEFAULT_ ERROR (Default Error Process.)
RCS	RCS version	RCS version	Text	240	RCS version
FROM_ROLE	From Role	From Role	Role		Item Attribute which holds the From Role. (Message #FROM_ ROLE)

ENGCLACT Specific Item Attributes (required attributes when you pass the change line ID to StartWorfklow API)

Internal Name	Display Name	Description	Type	Length	Misc info (used for etc)
CHANGE_LINE_ID	Change Line Id	Engineering Change Line Unique Identifier	Number		Primary Key Values
LINE_NAME	Change Line Name	Engineering Change Line Name	Text	240	Token replacement for messages.
LINE_DESCRIPTION	Change Line Description	Change Line Description	Text	4000	Token replacement for messages.
LINE_SEQUENCE_NUMBER	Change Line Sequence Number	Change Line Sequence Number	Number		Token replacement for messages.
LINE_STATUS	Change Line Status	Change Line Status	Text	80	Token replacement for messages.
LINE_ASSIGNEE_NAME	Change Line Assignee Name	Change Line Assignee Name	Text	360	Token replacement for messages.
LINE_ASSIGNEE_COMPANY	Change Line Assignee Company	Change Line Assignee Company	Text	360	Token replacement for messages.
LINE_REVIEWERS_ROLE	Change Line Reviewer Role	Change Line Reviewer Role	Role		Temporary placeholders for notifications
LINE_ASSIGNEE_ROLE	Change Line Assignee Role	Change Line Assignee Role	Role		Temporary placeholders for notifications
LINE_ATTACHMENTS	Attachments	Attachments	Document		Temporary placeholders for notifications: Notification Detail Page Attachment LinkFND: entity===

ENGCRT Specific Item Attributes

Internal Name	Display Name	Description	Type	Length	Misc info (used for etc)
ROUTE_ID	Route Id (Workflow Routing Id)	Route Id	Number		Primary Key Values for Change Route
STEP_ID	Current Step Id (Workflow Routing Step Id)	Current Step Id	Number		Primary Key Values for Current Step Id
ROUTE_PEO PLE_ROLE	Route (Workflow Routing) People Role	All people in the Workflow Routing workflow that have already been notified (i.e. completed or in process steps)	Role		Temporary placeholders for notifications

ENGCSTEP Specific Item Attributes

Internal Name	Display Name	Description	Type	Length	Misc info (used for etc)
ROUTE_ID	Workflow Routing Id	Workflow Routing Id	Number		Primary Key Values for Change Route
STEP_ID	Workflow Routing Step Id	Current Step Id	Number		Primary Key Values for Step Id
ROUTE_PEOPLE_ROLE	Workflow Routing People Role	All people in the Workflow Routing workflow that have already been notified (i.e. completed or in process steps)	Role		Temporary placeholders for notifications
STEP_SEQ_NUM	Workflow Routing Step Sequence Number	Workflow Routing Step Sequence Number	Number		Temporary placeholders for notifications
STEP_PEOPLE_ROLE	Workflow Routing Step People Role	Workflow Routing Step People Role	Role		Temporary placeholders for notifications
STEP_CONDITION	Workflow Routing Step Activity Condition	Workflow Routing Step Activity Condition	Text	30	Temporary placeholders for notifications
STEP_VOTING_OPTION	Workflow Routing Step Voting Option for Activity Condition	Workflow Routing Step Voting Option for Activity Condition	Lookup		Standard Voting Option
YES_RESPONSE_PERCENT	Yes Response Percentage of Votes	Yes Response Percentage of Votes	Number		Temporary placeholders for notifications
NO_RESPONSE_PERCENT	No Response Percentage of Votes	No Response Percentage of Votes	Number		Temporary placeholders for notifications

Change Management Workflow PL/SQL Packages

Change Management Workflow Private APIs will contain the following list of packages. These are private APIs for reference purposes only for your customization.

Package Name	File Name	Description	Location of Flat File Store on Server
ENG_WORKFLOW_PUB	ENGBWKFS.pls ENGBWKFB.pls	Private APIs: for developers or consultants at Oracle: This package contains All PL/SQL Function for Workflow Function Activity.	$ENG_TOP/patch/115/sql
ENG_WORKFLOW_UTIL	ENGUWKFS.pls ENGUWKFS.pls	Private APIs: This package contains utility functions and procedures for Eng Workflows	$ENG_TOP/patch/115/sql
ENG_WORKFLOW_NTF_UTIL	ENGUNTFS.pls ENGUNTFS.pls	This package contains utility functions and procedures for Eng Workflow Notifications	$ENG_TOP/patch/115/sql
ENG_WORKLFOW_EXT	ENGXWKFS.pls ENGXWKFB.pls	This package contains procedures that serve as "hooks" into the Eng Workflows for customization of the workflows. The "StartCustom-Workflow " and the "AbortCustomWork-flow" are called from ENG_WORKFLOW_UTIL package when a workflow process is started and aborted, re-spectively. Users with customized workflows can add their custom logic in these two pro-cedures to be executed at the appropriate times. Also GetCustomMes-sageBody is called from the seeded PL/SQL Document APIs when system generates notification. Users can customize the PL/SQL procedure to modify notification contents. Additionally, users can create new procedures in this package for custom workflow activities.	$ENG_TOP/patch/115/sql

B

Reading Data for User-Defined Attributes

Overview

You can use Advanced Product Catalog to define an unlimited number of attributes for items without programming them. You can define attributes using the setup user interface; definitions are stored in metadata tables. However, you may also need to read attribute data to interface with third party or custom systems.

Generate database views to read User-defined attributes data

Using the attribute group setup screens, you can generate database views for reading user-defined attributes data. These views hide the complexity of dealing with the attribute meta-data.

Using PL/SQL to Access Data for Attributes

Another way to access user-defined attribute data for a particular item is to use PL/SQL. The EGO_USER_ATTRS_DATA_PUB procedure follows:

PROCEDURE Get_User_Attrs_Data

p_api_version	IN NUMBER
,p_object_name	IN VARCHAR2
,p_pk_column_name_value_pairs	IN EGO_COL_NAME_VALUE_PAIR_ARRAY
,p_attr_group_request_table	IN EGO_ATTR_GROUP_REQUEST_TABLE
,p_user_privileges_on_object	IN EGO_VARCHAR_TBL_TYPE DEFAULT NULL
,p_entity_id	IN VARCHAR2 DEFAULT NULL
,p_entity_index	IN NUMBER DEFAULT NULL
,p_entity_code	IN VARCHAR2 DEFAULT NULL
,p_debug_level	IN NUMBER DEFAULT 0
,p_init_error_handler	IN VARCHAR2 DEFAULT FND_API.G_FALSE
,p_init_fnd_msg_list	IN VARCHAR2 DEFAULT FND_API.G_FALSE
,p_add_errors_to_fnd_stack	IN VARCHAR2 DEFAULT FND_API.G_FALSE
,p_commit	IN VARCHAR2 DEFAULT FND_API.G_FALSE
,x_attributes_row_table	OUT NOCOPY EGO_USER_ATTR_ROW_TABLE
,x_attributes_data_table	OUT NOCOPY EGO_USER_ATTR_DATA_TABLE
,x_return_status	OUT NOCOPY VARCHAR2
,x_errorcode	OUT NOCOPY NUMBER
,x_msg_count	OUT NOCOPY NUMBER
,x_msg_data	OUT NOCOPY VARCHAR2

p_api_version

Callers should pass a constant value of 1.0.

p_object_name

Callers should pass a constant value of 'EGO_ITEM'

p_pk_column_name_value_pairs

This is a table of EGO_COL_NAME_VALUE_PAIR_OBJ objects specifying the Inventory Item ID and Organization ID for which the caller wants attribute data. Callers should create the table with code like the following:

l_pk_column_values EGO_COL_NAME_VALUE_PAIR_ARRAY;

l_pk_column_values := EGO_COL_NAME_VALUE_PAIR_ARRAY(

EGO_COL_NAME_VALUE_PAIR_OBJ('INVENTORY_ITEM_ID', <your inventory item ID>) ,EGO_COL_NAME_VALUE_PAIR_OBJ('ORGANIZATION_ID', <your organization ID>));

p_attr_group_request_table

This is a table of EGO_ATTR_GROUP_REQUEST_OBJ objects, one for each attribute group whose values the caller wants. The objects also support requests for only some of the attributes in the attribute group. Each EGO_ATTR_GROUP_REQUEST_OBJ object in the table will look like the following:

EGO_ATTR_GROUP_REQUEST_OBJ(

NULL,

431,

'EGO_ITEMMGMT_GROUP',

<your Attribute Group internal name>,

<your Revision ID if the Attribute Group is associated at the Revision level; otherwise NULL>,

NULL,

NULL

<an optional comma-delimited list of the Attribute internal names whose values you want, or NULL>);

p_user_privileges_on_object

This is an optional table of VARCHAR2s, one for each privilege the caller wants validated against the View privileges of the attribute groups being requested. For example, if a user invokes a procedure that calls Get_User_Attrs_Data, and the calling procedure wants to enforce attribute group data security, the calling procedure could pass a list of the user's privileges based on his/her role on the current item. If this parameter is non-null, Get_User_Attrs_Data will enforce data security for those attribute groups that have a View privilege defined.

p_entity_id, p_entity_index, p_entity_code

Parameters for use with ERROR_HANDLER package. Can safely be defaulted unless caller wants to distinguish among errors for multiple items; if so, please investigate the ERROR_HANDLER package for more details.

p_debug_level

Parameter to control debug logging; can safely be defaulted.

p_init_error_handler, p_init_fnd_msg_list, p_add_errors_to_fnd_stack

More parameters for use with ERROR_HANDLER package; can safely be defaulted.

p_commit

Controls whether or not Get_User_Attrs_Data issues a COMMIT statement at the end of processing; present to comply with API standards, but since Get_User_Attrs_Data does not perform any DML, can safely be defaulted.

x_attributes_row_table

This is a table of EGO_USER_ATTR_ROW_OBJ objects, each one representing a row of user-defined attributes data (for example, one database table row of data). It contains the Extension ID value from EGO_MTL_SY_ITEMS_EXT_VL (stored as ROW_IDENTIFIER) and serves mainly as a way to organize the objects in x_attributes_data_table.

x_attributes_data_table

This is a table of EGO_USER_ATTR_DATA_OBJ objects, each one representing a single attribute value. All of the attribute values for a given row of data will share the same ROW_IDENTIFIER value; as with x_attributes_row_table, the ROW_IDENTIFIER value is the Extension ID from EGO_MTL_SY_ITEMS_EXT_VL. Each object also contains the attribute internal name (stored as ATTR_NAME) and its value (stored as ATTR_DISP_VALUE).

x_return_status, x_errorcode, x_msg_count, x_msg_data

Four standard OUT parameters indicating what happened in processing. The first, x_return_status, will equal FND_API.G_RET_STS_SUCCESS if processing succeeded, FND_API.G_RET_STS_ERROR if an expected error case was encountered, or FND_API.G_RET_STS_UNEXP_ERROR if something went wrong during processing. X_errorcode is not used. X_msg_count indicates how many error messages were logged with ERROR_HANDLER; if x_msg_count is 1, then x_msg_data contains that message (to save the caller the inconvenience of interacting with ERROR_HANDLER for only one message).

C

Synchronizing Item User-Defined Attributes with Item Descriptive Elements

Overview

You can define an unlimited number of user-defined attributes for items without programming. These attributes can have different data types, value sets, and can be parametrically searched. This appendix describes how to synchronize user-defined attribute values with descriptive element values. Oracle Advanced Product Catalog provides a public API to update Item Descriptive Element values. It also raises a business event when any pre-selected user-defined attribute values change. You can subscribe to this event and then call the Item Descriptive Element API to synchronize the descriptive elements with the attribute values. Note: If you want to enable business events for item user-defined attributes, then you must apply patch 3250729. A business event is raised when any pre-selected item user-defined attribute values change. This patch also provides a public API to maintain Descriptive Elements for a given item. A working knowledge of the Business Event System is necessary to understand this document and follow its instructions (e.g., enabling and then subscribing to the Business Event, writing a Rule Function).

"Event Manager" section of Chapter 8 and the "Standard API for an Event Rule Function" section of Chapter 6 in the Oracle Workflow Developer's Guide, also refer to the Oracle Workflow API Reference, as necessary.

User-Defined Attributes Business Event

In Oracle Advanced Product Catalog, you can configure user-defined attribute groups to raise a Business Event every time a row is added, altered, or deleted The name of the Business Event raised is

```
oracle.apps.ego.item.attributesChanged
```

The parameters passed by this Business Event are

```
                INVENTORY_ITEM_ID
                ORGANIZATION_ID
                EXTENSION_ID
                ATTR_GROUP_NAME
```
The Event is disabled by default and must be explicitly enabled for synchronization to work.

Item Descriptive Elements API

Oracle Advanced Product Catalog provides the following API (with accompanying data type) to maintain Descriptive Elements for a given item.

The data type used by the API follows:

```
ITEM_DESC_ELEMENT_TABLE: this is an associative array (also known
 as a PL/SQL table or an index-by table) of ITEM_DESC_ELEMENT reco
rds:
ITEM_DESC_ELEMENT IS RECORD

(
      ELEMENT_NAME          VARCHAR2(30)
      ,ELEMENT_VALUE         VARCHAR2(30)
      ,DESCRIPTION_DEFAULT  VARCHAR2(1)
     );

ELEMENT_NAME: Column ELEMENT_NAME from              MTL_DESCRIPTIVE
_ELEMENTS
ELEMENT_VALUE: Column ELEMENT_VALUE from MTL_DESCR_ELEMENT_VALUES
DESCRIPTION_DEFAULT: Column DEFAULT_ELEMENT_FLAG from MTL_DESCR_EL
EMENT_VALUES (indicates whether this Descriptive Element will be u
sed in the description generation)
```

The API signature is:

```
INV_ITEM_CATALOG_ELEM_PUB.Process_Item_Descr_Elements
(
 p_api_version        IN NUMBER
,p_init_msg_list      IN VARCHAR2 DEFAULT fnd_api.g_FALSE
,p_commit_flag        IN VARCHAR2 DEFAULT fnd_api.g_FALSE
,p_validation_level   IN NUMBER    DEFAULT
INV_ITEM_CATALOG_ELEM_PUB.g_VALIDATE_ALL
,p_inventory_item_id  IN NUMBER    DEFAULT  -999
,p_item_number        IN VARCHAR2 DEFAULT  NULL
,p_item_desc_element_table IN ITEM_DESC_ELEMENT_TABLE
,x_generated_descr    OUT VARCHAR2
,x_return_status      OUT VARCHAR2
,x_msg_count          OUT NUMBER
,x_msg_data           OUT VARCHAR2
);
```

p_api_version

Pass the value 1.0 for this parameter.

p_init_msg_list

This parameter can safely be defaulted.

p_commit_flag

Indicates whether or not the API commits the transaction.

p_validation_level

Determines how much validation occurs; can safely be defaulted.

p_inventory_item_id

Pass the value 1.0 for this parameter.

p_item_number

This parameter can safely be defaulted.

p_item_desc_element_table

Data type defined above.

x_generated_descr

Returns the Item Description generated from the updated Descriptive Elements. If you want to update the Item Description with this value, use Item Open Interface.

x_return_status

Returns one of the following:

```
    FND_API.G_RET_STS_SUCCESS, FND_API.G_RET_STS_ERROR, FND_API.G
_RET_STS_UNEXP_ERROR
```

x_msg_count

Number of messages logged by the API.

x_msg_data

The message logged if exactly one message was logged.

Subscribing to the Event and Calling the API

As briefly mentioned earlier, the Business Event oracle.apps.ego.item.attributesChanged must be explicitly enabled through the graphical user interface in order for synchronization to work. After this step is complete, you must subscribe to the User-Defined Attributes Business Event. Instructions for both of these steps are available in the "Event Manager" section of Chapter 8 of the Oracle Workflow Developer's Guide (hint: when subscribing, specify the following subscription's Event Filter:

```
        oracle.apps.ego.item.attributesChanged
```

As part of Event Subscription, define an Event Subscription Rule Function that will be called when the Event is raised. For information about how to write such a Rule Function, see the "Standard API for an Event Rule Function" section of Chapter 6 in the Oracle Workflow Developer's Guide (referring to the Oracle Workflow API Reference for details as necessary); also refer to the Sample Rule Function below.

This Rule Function should:

- Identify the attribute group that raised the Event and determine the correct Descriptive Elements for that attribute group (perhaps through customer-supplied mapping metadata).

- Fetch the data altered by the Event (see the following example Rule Function and also the White Paper "Reading Data for Item User-Defined Attributes" for more details) and process it as necessary.

- Call the Item Descriptive Elements API that is passing the processed data.

Sample Rule Function

```
FUNCTION Synch_Attrs_With_Desc_Elems (
        p_subscription_guid  IN     RAW
       ,p_event             IN OUT NOCOPY WF_EVENT_T
```

```
)
RETURN VARCHAR2
IS

TYPE ATTR_TO_DESC_ELEM_MAPPING_REC IS RECORD
    (
      ATTR_NAME              VARCHAR2(30)
     ,DESC_ELEM_NAME         VARCHAR2(30)
     ,DESCRIPTION_DEFAULT     VARCHAR2(1)
    );
    TYPE ATTR_TO_DESC_ELEM_MAPPING_TBL IS TABLE OF ATTR_TO_DESC_EL
EM_MAPPING_REC
      INDEX BY BINARY_INTEGER;

    l_parameter_list        WF_PARAMETER_LIST_T;
    l_next_parameter        WF_PARAMETER_T;
    l_dml_type_param        WF_PARAMETER_T;
    l_attr_group_name_param  WF_PARAMETER_T;
    l_inventory_item_id_param WF_PARAMETER_T;
    l_organization_id_param  WF_PARAMETER_T;
    l_revision_id_param      WF_PARAMETER_T;
    l_pk_column_values       EGO_COL_NAME_VALUE_PAIR_ARRAY;
    l_request_table          EGO_ATTR_GROUP_REQUEST_TABLE;
    x_attributes_row_table   EGO_USER_ATTR_ROW_TABLE;
    x_attributes_data_table  EGO_USER_ATTR_DATA_TABLE;
    x_return_status          VARCHAR2(1);
    x_errorcode              NUMBER;
    x_msg_count              NUMBER;
    x_msg_data               VARCHAR2(1000);
    l_current_mapping_rec    ATTR_TO_DESC_ELEM_MAPPING_REC;
    l_mapping_rec_table      ATTR_TO_DESC_ELEM_MAPPING_TBL;
    l_attributes_data_index  NUMBER;

    l_current_row_obj        EGO_USER_ATTR_ROW_OBJ;
    l_current_data_obj       EGO_USER_ATTR_DATA_OBJ;
    l_current_item_desc_elem INV_ITEM_CATALOG_ELEM_PUB.ITEM_DESC_E
LEMENT;

    l_item_desc_elem_table   INV_ITEM_CATALOG_ELEM_PUB.ITEM_DESC_E
LEMENT_TABLE;

    x_generated_descr        VARCHAR2(240);

    l_return_status          VARCHAR2(30) := 'SUCCESS';

BEGIN

    l_parameter_list := p_event.getParameterList();

    ----------------------------------------------------------
    -- Sort the parameters that the Business Event passed
    ----------------------------------------------------------
    FOR i IN l_parameter_list.FIRST .. l_parameter_list.LAST
    LOOP

    l_next_parameter := l_parameter_list(i);
    IF (l_next_parameter.getName() = 'DML_TYPE') THEN
        l_dml_type_param := l_next_parameter;
```

```
      ELSIF (l_next_parameter.getName() = 'ATTR_GROUP_NAME') THEN
          l_attr_group_name_param := l_next_parameter;
      ELSIF (l_next_parameter.getName() = 'INVENTORY_ITEM_ID') THEN
          l_inventory_item_id_param := l_next_parameter;
      ELSIF (l_next_parameter.getName() = 'ORGANIZATION_ID') THEN
          l_organization_id_param := l_next_parameter;
      ELSIF (l_next_parameter.getName() = 'REVISION_ID') THEN
          l_revision_id_param := l_next_parameter;
        END IF;

END LOOP;

 --------------------------------------------------------
 Copy the PK values into our EGO_COL_NAME_VALUE_PAIR_ARRAY object
 --------------------------------------------------------

   l_pk_column_values :=

            EGO_COL_NAME_VALUE_PAIR_ARRAY(
                EGO_COL_NAME_VALUE_PAIR_OBJ('INVENTORY_ITEM_ID',
                  l_inventory_item_id_param.getValue())
                  ,EGO_COL_NAME_VALUE_PAIR_OBJ('ORGANIZATION_ID',
                  l_organization_id_param.getValue())
                   );
    -----------------------------------------------------------------
 Prepare an EGO_ATTR_GROUP_REQUEST_TABLE object with one element,
an
 EGO_ATTR_GROUP_REQUEST_OBJ for the Attr Group that raised the Eve
nt
 -----------------------------------------------------------------

    l_request_table := EGO_ATTR_GROUP_REQUEST_TABLE();
    l_request_table.EXTEND();
    l_request_table(l_request_table.LAST) :=
                            EGO_ATTR_GROUP_REQUEST_OBJ(
                            NULL      --ATTR_GROUP_ID
                            ,431
                            ,'EGO_ITEMMGMT_GROUP'
                            ,l_attr_group_name_param.getValue()
                            ,l_revision_id_param.getValue()
                            ,NULL    --DATA_LEVEL_2
                            ,NULL    --DATA_LEVEL_3
                            ,NULL    --ATTR_NAME_LIST
                             );

    ----------------------------------------------------
    Get the data for this Event, if there is any
    ----------------------------------------------------
    IF (l_dml_type_param.getValue() <> 'DELETE') THEN

      EGO_USER_ATTRS_DATA_PUB.Get_User_Attrs_Data(
        p_api_version                 => 1.0
        ,p_object_name                => 'EGO_ITEM'
        ,p_pk_column_name_value_pairs => l_pk_column_values
        ,p_attr_group_request_table   => l_request_table
        ,p_user_privileges_on_object  => NULL
        ,p_entity_id                  => NULL
        ,p_entity_index               => NULL
```

```
            ,p_entity_code                  => NULL
            ,p_debug_level                  => 0
            ,p_init_error_handler           => FND_API.G_FALSE
            ,p_init_fnd_msg_list            => FND_API.G_FALSE
            ,p_add_errors_to_fnd_stack      => FND_API.G_FALSE
            ,p_commit                       => FND_API.G_FALSE
            ,x_attributes_row_table         => x_attributes_row_table
            ,x_attributes_data_table        => x_attributes_data_table
            ,x_return_status                => x_return_status
            ,x_errorcode                    => x_errorcode
            ,x_msg_count                    => x_msg_count
            ,x_msg_data                     => x_msg_data
        );


        ------------------------------------------------------------
        If the call failed, we report that an error occurred
        (in a production situation, we would retrieve and pass
        on all errors on the ERROR_HANDLER message stack)
        ------------------------------------------------------------
        IF (x_return_status IS NULL OR
            x_return_status <> FND_API.G_RET_STS_SUCCESS) THEN
          IF (x_return_status = FND_API.G_RET_STS_ERROR) THEN
            RAISE FND_API.G_EXC_ERROR;
          ELSE
            RAISE FND_API.G_EXC_UNEXPECTED_ERROR;
          END IF;
        END IF;
    END IF;

 --------------------------------------------------
 Process the Descriptive Elements according to the
 Attr Group that raised the Event and the DML type
 --------------------------------------------------
 IF (l_attr_group_name_param.getValue() = 'Pella_Windows') THEN

------------------------------------------------
Find mappings from Attr Group Attrs to Desc Elems; in this example
,
hard code the mapping from Attr Group Attrs to Desc Elems, but in
a
production situation customers may have some stored mapping betwee
n
the Attrs in an Attr Group associated to a Catalog Category and th
e
Descriptive Elements for that Catalog Category (similarly, in this
example we always set DESCRIPTION_DEFAULT as 'Y' for simplicity)
------------------------------------------------
l_current_mapping_rec.ATTR_NAME := 'Style';
l_current_mapping_rec.DESC_ELEM_NAME := 'Style';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'GB_Options';
l_current_mapping_rec.DESC_ELEM_NAME := 'Grille Bar Options';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;
```

```
l_current_mapping_rec.ATTR_NAME := 'Wood_Type';
l_current_mapping_rec.DESC_ELEM_NAME := 'Wood Type';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'Frame_Area';
l_current_mapping_rec.DESC_ELEM_NAME := 'Frame Area';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'GRADE';
l_current_mapping_rec.DESC_ELEM_NAME := 'Grade';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'PW_Exterior';
l_current_mapping_rec.DESC_ELEM_NAME := 'Primed Wood Exterior';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'DATE';
l_current_mapping_rec.DESC_ELEM_NAME := 'Date';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

ELSIF (l_attr_group_name_param.getValue() = 'WD_Attributes') THEN

l_current_mapping_rec.ATTR_NAME := 'STYLE';
l_current_mapping_rec.DESC_ELEM_NAME := 'WD Style';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'WD_Grade';
l_current_mapping_rec.DESC_ELEM_NAME := 'WD Grade';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'DATETIME';
l_current_mapping_rec.DESC_ELEM_NAME := 'Date';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

l_current_mapping_rec.ATTR_NAME := 'URL_link';
l_current_mapping_rec.DESC_ELEM_NAME := 'URL';
l_current_mapping_rec.DESCRIPTION_DEFAULT := 'Y';
l_mapping_rec_table(l_mapping_rec_table.COUNT + 1) := l_current_ma
pping_rec;

    END IF;
```

```
-------------------------------------------------------------------
 For every Desc Elem, get its value (a NULL value, or a DML_TYPE o
f DELETE,
 will result in a NULL Elem value), set its DESCRIPTION_DEFAULT fl
ag, and
 add it to the ITEM_DESC_ELEMENT_TABLE
 -------------------------------------------------------------------
 FOR i IN l_mapping_rec_table.FIRST .. l_mapping_rec_table.LAST
 LOOP

 l_current_mapping_rec := l_mapping_rec_table(i);
 l_current_item_desc_elem.ELEMENT_NAME := l_current_mapping_rec.DE
SC_ELEM_NAME;
 l_current_item_desc_elem.DESCRIPTION_DEFAULT := l_current_mapping
_rec.DESCRIPTION_DEFAULT;

 -------------------------------------------------------------------
-
This example uses single-row Attr Groups, so we don't need to make
 use
of the EGO_USER_ATTR_ROW_TABLE object; but for a multi-row Attr Gr
oup,
we would use the ROW_IDENTIFIER field of each EGO_USER_ATTR_ROW_OB
J to
find all the EGO_USER_ATTR_DATA_OBJ objects for that particular ro
w
 -------------------------------------------------------------------
 IF (x_attributes_row_table IS NOT NULL AND
     x_attributes_row_table.COUNT > 0 AND
     x_attributes_data_table IS NOT NULL AND
     x_attributes_data_table.COUNT > 0) THEN

  l_attributes_data_index := x_attributes_data_table.FIRST;
  WHILE l_attributes_data_index <= x_attributes_data_table.LAST
    LOOP
      EXIT WHEN l_current_item_desc_elem.ELEMENT_VALUE IS NOT NULL
;
        l_current_data_obj := x_attributes_data_table(l_attributes
_data_index);
        IF (l_current_data_obj.ATTR_NAME = l_current_mapping_rec.A
TTR_NAME) THEN
         l_current_item_desc_elem.ELEMENT_VALUE := SUBSTRB(l_curre
nt_data_obj.ATTR_DISP_VALUE, 1, 30);
           END IF;
        l_attributes_data_index := x_attributes_data_table.NEXT(l
_attributes_data_index);
          END LOOP;
      END IF;

 ---------------------------------------------------------
 -- Now we add the ITEM_DESC_ELEMENT record into our table
 ---------------------------------------------------------
l_item_desc_elem_table(l_item_desc_elem_table.COUNT + 1) := l_curr
ent_item_desc_elem;

    END LOOP;
 ----------------------------------------------------
```

```
      Now that we have all the necessary data and metadata, we
      call the API to update the Desc Elems for this Attr Group
      (note that the API returns the Item Description generated
      from the updated Descriptive Elements; if required, you
      can update the Description through Item Open Interface)
      ----------------------------------------------------
INV_ITEM_CATALOG_ELEM_PUB.Process_Item_Descr_Elements(
          p_api_version                => 1.0
         ,p_inventory_item_id          => l_inventory_item_id_param.
getValue()
         ,p_item_desc_element_table    => l_item_desc_elem_table
         ,x_generated_descr            => x_generated_descr
         ,x_return_status              => x_return_status
         ,x_msg_count                  => x_msg_count
         ,x_msg_data                   => x_msg_data
      );
      RETURN l_return_status;
EXCEPTION
  WHEN OTHERS THEN
    WF_CORE.CONTEXT('EGO_EF_QA_PUB', 'Synch_Attrs_With_Desc_Elems'
,
            p_event.getEventName(), p_subscription_guid);
    WF_EVENT.setErrorInfo(p_event, 'ERROR');
      RETURN 'ERROR';
END Synch_Attrs_With_Desc_Elems;
```