## Contents

### 1 Overview
- Marketing Offering: Overview ................................................................. 1-1
- Manage Application Implementation ......................................................... 1-3

### 2 Define Common Applications Configuration for Marketing
- Common Implementation: Overview ......................................................... 2-1
- Define Synchronization of Users and Roles from LDAP .......................... 2-2
- Define Implementation Users ................................................................. 2-2
- Define Currencies and Currency Rates .................................................. 2-4
- Euro Currency Derivation: Explained .................................................. 2-5
- FAQs for Currencies and Currency Rates ............................................. 2-5
- Define Enterprise Structures for Marketing ............................................. 2-10
- Define Social Networking ................................................................. 2-92
- Define Workforce Profiles ................................................................. 2-92
- Define Security for Customer Relationship Management ...................... 2-119
- Define Automated Governance, Risk, and Performance Controls .......... 2-164
- Define Approval Management for Customer Relationship Management .... 2-173
- Define Help Configuration ................................................................. 2-174
- Define Application Toolkit Configuration ............................................. 2-186
- Maintain Common Reference Objects .............................................. 2-189
- Define WebLogic Communication Services Configuration .................. 2-287

### 3 Define Common CRM Configuration
- Define Source Systems ................................................................. 3-1
- Define Identifiers ................................................................................. 3-2
- Define Classifications ........................................................................ 3-3
- Define Resource Information ............................................................ 3-4
- Define Resource Organization Information ......................................... 3-7
- Define Resource Role Information ..................................................... 3-8
- Define Products: Define Basic Items ................................................... 3-10
- Define Products: Define Advanced Items ........................................... 3-18
- Define Sales Catalogs ........................................................................ 3-22
- Define File-Based Data Import .......................................................... 3-35
- Define Data Export ............................................................................. 3-57
- Manage Calendar Profile Option ...................................................... 3-63
- Define Sales Prediction Configuration ............................................... 3-64
- Define Outlook Integration ............................................................ 3-65
- Set Up Customer Center ................................................................... 3-75

### 4 Define Territory Management Configuration
5 Define Lead Management

How Lead Components Fit Together .................................................................5-1
Configure Assignment Manager for Lead Processing .................................5-3
Configure Assessment Reference Data for Sales Leads ...............................5-29
Configure Notes for Sales Leads .................................................................5-38
Configure Tasks for Sales Leads ...............................................................5-39
Configure Sales Leads ..............................................................................5-41

6 Define Common Marketing Configuration

Define Integration Points for Segmentation Manager .......................................6-1
Manage Marketing Standard URLs ..............................................................6-1
Review Diagnostics for Marketing ................................................................6-2
Define Marketing Promotions ....................................................................6-3
Define Campaign and Response Notes for Marketing .................................6-4
Define Campaign Tasks for Marketing .......................................................6-5
Define Marketing Templates: Manage Marketing Campaign Templates ..........6-7
Define Marketing Templates: Manage Marketing Treatment Templates ..........6-15

7 Define Segmentation Manager

Marketing Segmentation: Highlights ...........................................................7-1

8 Define Transactional Business Intelligence Configuration

Define Transactional Business Intelligence Configuration: Highlights ..........8-1
Access to Person Data ................................................................................8-1

9 Define Extensions for Marketing

Define Custom Enterprise Scheduler Jobs ...................................................9-1
Customization and Sandboxes ..................................................................9-2

10 Importing and Exporting Setup Data

Configuration Packages: Explained ...........................................................10-1
Exporting and Importing Setup Data: Explained .........................................10-1
Moving Common Reference Objects .........................................................10-2
Preface

This Preface introduces the guides, online help, and other information sources available to help you more effectively use Oracle Fusion Applications.

Oracle Fusion Applications Help

You can access Oracle Fusion Applications Help for the current page, section, activity, or task by clicking the help icon. The following figure depicts the help icon.

With a local installation of help, you can add custom help files to replace or supplement the provided content. Help content patches are regularly made available to ensure you have access to the latest information. Patching does not affect your custom content.

Oracle Fusion Applications Guides

Oracle Fusion Applications guides are a structured collection of the help topics, examples, and FAQs from the help system packaged for easy download and offline reference, and sequenced to facilitate learning. You can access the guides from the Guides menu in the global area at the top of Oracle Fusion Applications Help pages.

Note
The Guides menu also provides access to the business process models on which Oracle Fusion Applications is based.

Guides are designed for specific audiences:

- **User Guides** address the tasks in one or more business processes. They are intended for users who perform these tasks, and managers looking for an overview of the business processes. They are organized by the business process activities and tasks.

- **Implementation Guides** address the tasks required to set up an offering, or selected features of an offering. They are intended for implementors. They are organized to follow the task list sequence of the offerings, as displayed within the Setup and Maintenance work area provided by Oracle Fusion Functional Setup Manager.

- **Concept Guides** explain the key concepts and decisions for a specific area of functionality. They are intended for decision makers, such as chief financial officers, financial analysts, and implementation consultants. They are organized by the logical flow of features and functions.
• Security Reference Manuals describe the predefined data that is included in the security reference implementation for one offering. They are intended for implementors, security administrators, and auditors. They are organized by role.

To supplement these guides, which cover specific business processes and offerings, the following guides address common areas:

Other Information Sources

My Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Use the My Oracle Support Knowledge Browser to find documents for a product area. You can search for release-specific information, such as patches, alerts, white papers, and troubleshooting tips. Other services include health checks, guided lifecycle advice, and direct contact with industry experts through the My Oracle Support Community.

Oracle Enterprise Repository for Oracle Fusion Applications

Oracle Enterprise Repository for Oracle Fusion Applications provides visibility into service-oriented architecture assets to help you manage the lifecycle of your software from planning through implementation, testing, production, and changes. In Oracle Fusion Applications, you can use the Oracle Enterprise Repository for Oracle Fusion Applications for:

• Technical information about integrating with other applications, including services, operations, composites, events, and integration tables. The classification scheme shows the scenarios in which you use the assets, and includes diagrams, schematics, and links to other technical documentation.

• Publishing other technical information such as reusable components, policies, architecture diagrams, and topology diagrams.

The Oracle Fusion Applications information is provided as a solution pack that you can upload to your own deployment of Oracle Enterprise Repository for Oracle Fusion Applications. You can document and govern integration interface assets provided by Oracle with other assets in your environment in a common repository.

Documentation Accessibility

For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/us/corporate/accessibility/index.html.
Comments and Suggestions

Your comments are important to us. We encourage you to send us feedback about Oracle Fusion Applications Help and guides. Please send your suggestions to oracle_fusion_applications_help_ww@oracle.com. You can use the Send Feedback to Oracle link in the footer of Oracle Fusion Applications Help.
Marketing Offering: Overview

Using the marketing business process area, your enterprise can create consumer awareness of your products or services. You can send marketing messages directly to consumers. Drive prospects towards well qualified leads for your organization's sales force. Define your company's marketing strategy, create marketing plans, define pricing and budgets, identify customer segments, create marketing collateral, and execute campaigns across multiple channels. Integration across marketing, planning, pricing, campaign development and execution, and lead management enables marketing effectiveness, intelligence, and performance analysis. You can effectively close the loop in your marketing campaigns.

Before you begin, use the Getting Started page in the Setup and Maintenance work area to access reports for each offering, including full lists of setup tasks, descriptions of the options and features you can select when you configure the offering, and lists of business objects and enterprise applications associated with the offering.

The first implementation step is to configure the offerings in the Setup and Maintenance work area by selecting the offerings and options that you want to make available to implement. For the Marketing offering, you can select the following options:

- E-Mail Server for Marketing
- Lead Management
- Segmentation Server for Marketing
- Marketing Business Intelligence Analytics

Next, create one or more implementation projects for the offerings and options that you want to implement first, which generates task lists for each project. The application implementation manager can customize the task list and assign and track each task.

If you select all of the options, the generated task list for this offering contains the following groups of tasks:

- Define Common Applications Configuration
• Define Common CRM Configuration
• Define Territory Management Configuration
• Define Lead Management
• Define Common Marketing Configuration
• Define Campaign Fulfillment System
• Define E-Mail Server
• Define Segmentation Manager
• Define Transactional Business Intelligence Configuration
• Define Extensions for Marketing

**Define Common Applications Configuration**

Use this task list to manage definitions used across offerings, typically applying to multiple products and product families. These definitions include enterprise structures, workforce profiles, security, and approval rules, amongst others.

You can find other information that supports the common implementation tasks in the Oracle Fusion Applications Concepts Guide.

**Define Common CRM Configuration**

Use this task list to define and manage the setup for common options within the customer relationship management set of business processes.

**Define Territory Management Configuration**

Use this task list to define and manage the attributes, attribute values, metrics, policies, and measure information that is related to territory management.

**Define Lead Management**

Use this task list to define and manage the setup to support sale leads creation and followup functions.

**Define Common Marketing Configuration**

Use this task list to define and manage the setup for common functions within the marketing business process.

**Define Campaign Fulfillment System**

Use this task list to create and manage marketing suppliers, such as call centers and fulfillment companies, that provide campaign distribution services.

**Define E-Mail Server**

Use this task list to install and manage e-mail server configuration changes. The e-mail server provides the e-mail sending daemon that delivers e-mail, the bounce-handling daemon that tracks e-mails that cannot be delivered, and the click-through daemon that tracks e-mail recipient responses.
Define Segmentation Manager

Use this task list to define and manage Oracle Fusion Segmentation configuration, such as list formats, merge fields, and configuration parameters, for Oracle Fusion Marketing.

Define Transactional Business Intelligence Configuration

Use this task list to configure Oracle Transactional Business Intelligence for ad hoc reporting, including managing the repository, connections, presentation catalog, and currency type display.

Define Extensions for Marketing

Use this task list to define extensions such as custom Oracle Enterprise Scheduler jobs.

You can also customize and extend applications using other tools. For more information, see the Oracle Fusion Applications Extensibility Guide.

Manage Application Implementation

Manage Application Implementation: Overview

The Manage Applications Implementation business process enables rapid and efficient planning, configuration, implementation, deployment, and ongoing maintenance of Oracle Fusion applications through self-service administration.

The Setup and Maintenance work area offers you the following benefits:

- Prepackaged lists of implementation tasks
  Task lists can be easily configured and extended to better fit with business requirements. Auto-generated, sequential task lists include prerequisites and address dependencies to give full visibility to end-to-end setup requirements of Oracle Fusion applications.

- Rapid start
  Specific implementations can become templates to facilitate reuse and rapid-start of consistent Oracle Fusion applications setup across many instances.

- Comprehensive reporting
  A set of built-in reports helps to analyze, validate and audit configurations, implementations, and setup data of Oracle Fusion applications.

With Oracle Fusion Functional Setup Manager you can:

- Learn about and analyze implementation requirements.
- Configure Oracle Fusion applications to match your business needs.
• Get complete visibility to setup requirements through guided, sequential task lists downloadable into Excel for project planning.

• Enter setup data through easy-to-use user interfaces available directly from the task lists.

• Export and import data from one instance to another for rapid setup.

• Validate setup by reviewing setup data reports.

• Implement all Oracle Fusion applications through a standard and consistent process.

Implementation Projects: Explained

An implementation project is the list of setup tasks you need to complete to implement selected offerings and options. You create a project by selecting the offerings and options you want to implement together. You manage the project as a unit throughout the implementation lifecycle. You can assign these tasks to users and track their completion using the included project management tools.

Maintaining Setup Data

You can also create an implementation project to maintain the setup of specific business processes and activities. In this case, you select specific setup task lists and tasks.

Exporting and Importing

Implementation projects are also the foundation for setup export and import. You use them to identify which business objects, and consequently setup data, you will export or import and in which order.

Selecting Offerings

When creating an implementation project you see the list of offerings and options that are configured for implementation. Implementation managers specify which of those offerings and options to include in an implementation project. There are no hard and fast rules for how many offerings you should include in one implementation project. The implementation manager should decide based on how they plan to manage their implementations. For example, if you will implement and deploy different offerings at different times, then having separate implementation projects will make it easier to manage the implementation life cycles. Furthermore, the more offerings you included in an implementation project, the bigger the generated task list will be. This is because the implementation task list includes all setup tasks needed to implement all included offerings. Alternatively, segmenting into multiple implementation projects makes the process easier to manage.

Offerings: Explained

Offerings are application solution sets representing one or more business processes and activities that you typically provision and implement as a unit.
They are, therefore, the primary drivers of functional setup of Oracle Fusion applications. Some of the examples of offerings are Financials, Procurement, Sales, Marketing, Order Orchestration, and Workforce Deployment. An offering may have one or more options or feature choices.

**Implementation Task Lists**

The configuration of the offerings will determine how the list of setup tasks is generated during the implementation phase. Only the setup tasks needed to implement the selected offerings, options and features will be included in the task list, giving you a targeted, clutter-free task list necessary to meet your implementation requirements.

**Enabling Offerings**

Offerings and their options are presented in an expandable and collapsible hierarchy to facilitate progressive decision making when specifying whether or not an enterprise plans to implement them. An offering or its options can either be selected or not be selected for implementation. Implementation managers decide which offerings to enable.

**Provisioning Offerings**

The Provisioned column on the Configure Offerings page shows whether or not an offering is provisioned. While you are not prevented from configuring offerings that have not been provisioned, ultimately the users are not able to perform the tasks needed to enter setup data for those offerings until appropriate enterprise applications (Java EE applications) are provisioned and their location (end point URLs) is registered.

**Options: Explained**

Each offering in general includes a set of standard functionality and a set of optional modules, which are called options. For example, in addition to standard Opportunity Management, the Sales offering includes optional functionality such as Sales Catalog, Sales Forecasting, Sales Prediction Engine, and Outlook Integration. These optional functions may not be relevant to all application implementations. Because these are subprocesses within an offering, you do not always implement options that are not core to the standard transactions of the offering.

**Feature Choices: Explained**

Offerings include optional or alternative business rules or processes called feature choices. You make feature selections according to your business requirements to get the best fit with the offering. If the selected offerings and options have dependent features then those features are applicable when you implement the corresponding offering or option. In general, the features are set with a default configuration based on their typical usage in most implementations. However, you should always review the available feature choices for their selected offerings and options and configure them as appropriate for the implementation.
You can configure feature choices in three different ways:

**Yes or No**
If a feature can either be applicable or not be applicable to an implementation, a single checkbox is presented for selection. Check or uncheck to specify yes or no respectively.

**Single Select**
If a feature has multiple choices but only one can be applicable to an implementation, multiple choices are presented as radio buttons. You can turn on only one of those choices.

**Multi-Select**
If the feature has multiple choices but one or more can be applicable to an implementation then all choices are presented with a checkbox. Select all that apply by checking the appropriate choices.
Define Common Applications Configuration for Marketing

Common Implementation: Overview

Common implementation involves accessing tasks that are available in multiple offerings, or that apply to multiple products and product families. The Define Common Applications Configuration task list and other activities include these common setup and implementation tasks.

You can find other information in support of common implementation in the Oracle Fusion Applications Concepts Guide.

In addition, you can customize and extend applications using various tools. For more information, see the Oracle Fusion Applications Extensibility Guide.

Define Common Applications Configuration Task List

Use the Define Common Applications Configuration task list to set up and administer an implementation of behaviors across offerings.

Most Oracle Fusion Applications offerings include the Define Common Applications Configuration task list for implementing what is common in multiple or all Oracle Fusion applications. The task lists and tasks within Define Common Applications Configuration can be present in all offerings, some, or just a single offering.

Common implementation includes such tasks as setting up security, defining enterprise structures, configuring Oracle Fusion Applications Help, and setting options. Many of the common implementation tasks involve configuring reference objects provided by Oracle Fusion Middleware Extensions for Applications (Applications Core), such as messages, flexfields, document sequences, and profile options. Some common implementation tasks involve configuring features provided by Oracle Application Toolkit (ATK), such as the Watchlist. Other common implementation tasks involve Oracle Fusion Applications products such as the Assign Balancing Segment Values to Ledger task in Oracle Fusion General Ledger.

Other Common Setup and Maintenance Tasks

Other setup and maintenance tasks exist in multiple offerings but not in the Define Common Applications Configuration task list. Use these other task lists to define an Oracle Fusion Transactional Business Intelligence configuration, and to define extensions such as custom Oracle Enterprise Scheduler jobs.
You can access common implementation tasks and task lists by starting in the Setup and Maintenance Overview page and searching for task lists by name. Setup and Maintenance is available from the Administration menu to users provisioned with appropriate roles. The Administration menu provides access to other tasks, such as for customization.

**Define Synchronization of Users and Roles from LDAP**

**User and Role Synchronization: Explained**

Oracle Identity Management (OIM) maintains Lightweight Directory Access Protocol (LDAP) user accounts for users of Oracle Fusion applications. OIM also stores the definitions of abstract, job, and data roles and holds information about roles provisioned to users. During implementation, any existing information about users, roles, and roles provisioned to users must be copied from the LDAP directory to the Oracle Fusion Applications tables. Once the Oracle Fusion Applications tables are initialized with this information, it is maintained automatically. To perform the initialization, you run the process Retrieve Latest LDAP Changes.

**Define Implementation Users**

**Initial Security Administration: Critical Choices**

Initial security administration is performed by an administrative user who is created and provisioned with the IT Security Manager role.

---

**Note**

The Oracle Fusion Applications installation process creates a super user account, which is available for signing into Oracle Fusion Applications to create a user provisioned with the IT Security Manager role.

Initial security administration primarily establishes at least one implementation user. The IT security manager must provision the initial implementation user with sufficient access to set up the enterprise, including provisioning of the Application Implementation Consultant role to the implementation users.

Perform the following tasks to establish implementation users with appropriate access.

- Create implementation users.
- Create a data role for performing Human Capital Management (HCM) setup steps.
- Create a data role for implementation users.
- Provision roles to implementation users.

**Creating Implementation Users**

Create one or more implementation users by performing the Create Implementation Users task in Oracle Identity Management (OIM). An
implementation user must exist to set up the enterprise in Oracle Fusion Applications.

**Note**
User and user account information is stored in the Lightweight Directory Access Protocol (LDAP) store. The implementation user does not need to be associated with a person in Human Resources (HR).

In the security reference implementation, the IT Security Manager job role hierarchy includes the User Management Duty role, which is entitled to create and manage users (the entitlement is Manage User Principal). This entitlement provides the access necessary to perform the Create Implementation Users task in OIM.

**Creating a Data Role for Performing HCM Setup Steps**

No predefined roles exist in the Oracle Fusion Applications security reference implementation to access the data necessary for setting up the HCM structures of the enterprise.

Setting up the HCM structures includes the following.
- HR structures, such as establishing job and position structures
- Departments and organization trees
- Facilities and inventory organizations
- HCM security profiles

These setup tasks are commonly done by application implementation consultants with administrator access, for example an HCM Application Administrator View All data role.

**Note**
Administrator and implementation roles of the Oracle Fusion Applications security reference implementation are defined to access all other elements of the enterprise that need to be set up, such as the following.
- Reference data sharing
- Legal jurisdictions and authorities
- Legal entities
- Chart of accounts for enterprise structures
- Accounting configurations for enterprise structures
- Business units
- Project organizations

**Creating a Data Role for Implementation Users**

Create a Human Capital Management Application Administrator View All data role.

This data role is based on the Human Capital Management Application Administrator job role and extends that role with unrestricted access to data in the secured objects that the role is authorized to access. Users assigned to this data role can perform all of the HCM setup steps.
Caution

Once an implementation user with a View All data role has completed HCM security setup, it may be prudent to revoke the role and provision it only when specific HCM security setup changes are necessary. A View All data role grants broad access to all business units, reference data sets, and so on. Security setup in other offerings are not data security enabled and do not require a View All data role for enterprise setup.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Data Role Management Duty role, which is entitled to create a data role for Human Capital Management Application Administrator (the entitlement is Manage HCM Data Role). This entitlement provides the access necessary to perform the Create Data Role for Implementation Users task in Oracle Fusion Global Human Resources.

Provisioning Roles to Implementation Users

Provision the implementation user with one or more roles by performing the Provision Roles to Implementation Users task in Oracle Identity Manager (OIM).

For example, assign a role to the implementation user that provides the access necessary for setting up the enterprise, such as an HCM Application Administrator View All data role. Depending on the implementation, provision the predefined Applications Implementation Consultant role or a product family-specific administrator role, such as the predefined Financials Applications Administrator, to the implementation user. These predefined roles are available for selection in OIM.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Identity User Administrators and Role Administrators roles, which entitle you to provision users with roles. This entitlement provides the access necessary to perform the Provision Roles to Implementation Users task in OIM.

Define Currencies and Currency Rates

Defining Currencies: Points to Consider

When creating or editing currencies, consider these points relevant to entering the currency code, date range, or symbol for the currency.

Currency Codes

You cannot change a currency code after you enable the currency, even if you later disable that currency.

Date Ranges

Users can enter transactions denominated in the currency only for the dates within the specified range. If you do not enter a start date, then the currency is valid immediately. If you do not enter an end date, then the currency is valid indefinitely.
Symbols

Even if you enter a symbol for a currency, the symbol is not always displayed when an amount is displayed in this currency. Some applications use currency symbols when displaying amounts. Others, like Oracle Fusion General Ledger, do not.

Euro Currency Derivation: Explained

Use the Derivation Type, Derivation Factor, and Derivation Effective Date fields to define the relationship between the official currency (Euro) of the European Monetary Union (EMU) and the national currencies of EMU member states. For each EMU currency, you define its Euro-to-EMU fixed conversion rate and the effective starting date.

Note

If you need to use a different currency code for Euro, you can disable the predefined Euro currency and create a new one.

Derivation Type

The Euro currency derivation type is used only for the Euro, and the Euro derived derivation type identifies national currencies of EMU member states. All other currencies do not have derivation types.

Derivation Factor

The derivation factor is the fixed conversion rate by which you multiply one Euro to derive the equivalent EMU currency amount. The Euro currency itself should not have a derivation factor.

Derivation Effective Date

The derivation effective date is the date on which the relationship between the EMU currency and the Euro begins.

FAQs for Currencies and Currency Rates

When do I create or enable currencies?

Create currencies to use, for example for reporting purposes, if they are not already provided. All currencies from the International Organization for Standardization (ISO) 4217 standard are provided.

Enable any currency other than USD for use in Oracle Fusion Applications, for example for displaying monetary amounts, assigning to sets of books, entering transactions, and recording balances. Only USD is enabled by default.

What's the difference between precision, extended precision, and minimum accountable unit for a currency?

Precision is the number of digits to the right of the decimal point used in regular currency transactions. Extended precision is the number of digits to the right
of the decimal point used in calculations for this currency, and it must be greater than or equal to the standard precision. For example, USD would have 2 for precision because amounts are transacted as such, for example $1.00. For calculations, for example adding USD amounts, you might want the application to be more precise than two decimal digits, and would enter an extended precision accordingly.

**Note**

Some applications use extended precision. Others, such as Oracle Fusion General Ledger, do not.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for the cent. This unit does not necessarily correspond to the precision for all currencies.

**What's a statistical unit currency type?**

The statistical unit currency type is used only for the Statistical (STAT) currency. The Statistical currency is used to record statistics such as the number of items bought and sold. Statistical balances can be used directly in financial reports, allocation formulas, and other calculations.

**Manage Conversion Rate Types**

**Creating Conversion Rate Types: Critical Choices**

Maintain different conversion rates between currencies for the same period with the Oracle Fusion General Ledger conversion rate types functionality. Four predefined daily conversion rate types are seeded: Spot, Corporate, User, and Fixed, allowing you to use different rate types for different business needs. During journal entry, the conversion rate is provided automatically by the General Ledger based on the selected conversion rate type and currency, unless the rate type is user. For user rate types, you must enter the conversion rate. Define additional rate types as needed. Set your most frequently used rate type as the default. Conversion rate types cannot be deleted.

Assign conversion rate types to automatically populate the associated rate for your period average and period end rates for the ledger. For example, you can assign the predefined rate type **Spot** to populate your period average rates and the predefined rate type **Corporate** to populate your period end rates. Period average and period end rates are used in translation of account balances.

Conversion rate types are used to automatically assign a rate when you perform the following accounting functions:

- Convert foreign currency journal amounts to ledger currency equivalents
- Convert journal amounts from source ledgers to reporting currencies or secondary ledgers
- Run Revaluation or Translation processes

In creating new conversion rates, decide whether to do the following:

- Enforce inverse relationships
- Select pivot currencies
• Select contra currencies
• Enable cross rates and allow cross rate overrides
• Maintain cross rate rules

**Enforce Inverse Relationships**

Check the **Enforce Inverse Relationship** check box to specify whether or not to enforce the automatic calculation of inverse conversion rates when defining daily rates.

<table>
<thead>
<tr>
<th>Action</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checked</td>
<td>When you enter a daily rate to convert currency A to currency B, General Ledger automatically calculates the inverse rate, currency B to A, and enters it in the adjacent column. If either rate is changed, the application automatically recalculates the other rate. You can update the application calculated inverse rate, but once you do, the related rate is updated. The check box enforces that the inverse relationship is maintained but does not prevent you from updating the rates.</td>
</tr>
<tr>
<td>Unchecked</td>
<td>General Ledger calculates the inverse rate but you can change the rate and update the daily rates table without the corresponding rate being updated.</td>
</tr>
</tbody>
</table>

**Select Pivot Currencies**

Select a pivot currency that is commonly used in your currency conversions. A pivot currency is the central currency that interacts with contra currencies. For example, you set up a daily rate between the US dollar (USD) and the Euro currency (EUR) and another between the USD and the Canadian dollar (CAD). USD is the pivot currency in creating a rate between EUR and CAD. EUR and CAD are the contra currencies. Select the pivot currency from the list of values which contains those currencies that are enabled, effective, not a statistical (STAT) currency. The description of the pivot currency is populated automatically based on the currency definition.

If you want the application to create cross rates against a base currency, define the base currency as the pivot currency. Selected pivot currencies can be changed in the Rate Types page.

**Select Contra Currencies**

Select currencies available on the list of values as contra currencies. The available currencies are those currencies which are enabled, effective, not STAT currency, and not the pivot currency selected earlier. The description of the contra currency is populated automatically based on the currency definition. Add or delete contra currencies in the Contra Currencies region of the Rate Types page.

**Enable Cross Rates and Allow Cross Rate Overrides**

Check the **Enable Cross Rates** check box to calculate conversion rates based on defined currency rate relationships. General Ledger calculates cross rates.
based on your defined cross rate rules. Associate your cross rate rules with a conversion rate type, pivot currency, and contra currencies. Cross rates facilitate the creation of daily rates by automatically creating the rates between contra currencies based on their relationship to a pivot currency. If the **Enable Cross Rates** check box is changed to unchecked after entering contra currencies, the application stops calculating cross rates going forward for that particular rate type. All the earlier calculated cross rates for that rate type remain in the database unless you manually delete them.

For example, if you have daily rates defined for the pivot currency, USD to the contra currency, EUR, and USD to another contra currency, CAD, the application will automatically create the rates between EUR to CAD and CAD to EUR. This prevents the need to manually define the EUR to CAD and CAD to EUR rates.

Check the **Allow Cross Rates Override** check box to permit your users to override application generated cross rates. If you accept the default of unchecked, the application generated cross rates cannot be overridden.

**Maintain Cross Rate Rules**

Define or update your cross rate rules at any time by adding or removing contra currency assignments. Add a contra currency to a cross rate rule and run the Daily Rates Import and Calculation program to generate the new rates. If you remove a cross rate rule or a contra currency from a rule, any cross rates generated previously for that contra currency remain unless you manually delete them. Changes to the rule are not retroactive and will not affect previously stored cross rates. The Cross Rate program generates as many rates as possible and skips currencies where one component of the set is missing.

---

**Note**

With a defined web service that extracts daily currency conversion rates from external services, for example Reuters, currency conversion rates are automatically updated for the daily rates and all cross currency relationships.

---

**Using Rate Types: Examples**

There are four seeded conversion rate types in Oracle Fusion applications:

- Spot
- Corporate
- User
- Fixed

**Scenario**

You are the general ledger accountant for InFusion America Inc. You are entering a journal entry to capture three transactions that were transacted in three different foreign currencies:

- Canadian dollar (CAD): A very stable currency
- Mexican Peso (MXP): A fluctuating currency
- Hong Kong dollar (HKD): An infrequently used currency
You enter two lines with accounts and amounts for each foreign currency transaction. Based on your company procedures, you select the appropriate rate type to populate the rate for **Corporate** and **Spot** rate types from your daily rates table. You manually enter the current rate for the **User** rate type.

<table>
<thead>
<tr>
<th>Currency Selected</th>
<th>Rate Type Selected</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD</td>
<td>Corporate</td>
<td>Entered a periodic type of transaction. Your company has established a daily rate to use for the entire month across divisions for all transactions in CAD. CAD is a stable currency that only fluctuates slightly over the month.</td>
</tr>
<tr>
<td>MXP</td>
<td>Spot</td>
<td>Entered a periodic type of transaction. Your company enters daily rates each day for MXP because this currency is unstable and fluctuates.</td>
</tr>
<tr>
<td>HKD</td>
<td>User</td>
<td>Entered a one time transaction. Your company does not maintain daily rates in HKD.</td>
</tr>
</tbody>
</table>

**Note**

Your company does not currently use the **Fixed** rate type. From January 1, 1999, the conversion rate of the French franc (FRF) against the euro currency (EUR) was set at a fixed rate of 1 EUR to 6.55957 FRF. Your French operations were started in 2007, so you maintain all your French business records in the EUR.

**FAQs for Manage Conversion Rate Types**

**What's the difference between spot, corporate, user, and fixed rate types?**

Spot, corporate, user, and fixed conversion rate types differ based on the fluctuations of your entered foreign currency and your company procedures for maintaining daily rates.

<table>
<thead>
<tr>
<th>Rate Type</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spot</td>
<td>For currencies with fluctuating conversion rates or when exact currency conversion is needed.</td>
</tr>
<tr>
<td>Corporate</td>
<td>For establishment of a standard rate across your organization for a stable currency.</td>
</tr>
<tr>
<td>User</td>
<td>For infrequent entries where your daily rates for the entered foreign currency are not set up.</td>
</tr>
<tr>
<td>Fixed</td>
<td>For rates where the conversion is constant between two currencies.</td>
</tr>
</tbody>
</table>

If you have infrequent foreign currency transactions, the user rate type can simplify your currency maintenance while providing an accurate conversion rate on the date of the transaction.
Define Enterprise Structures for Marketing

Enterprise Structures: Overview

Oracle Fusion Applications have been designed to ensure your enterprise can be modeled to meet legal and management objectives. The decisions about your implementation of Oracle Fusion Applications are affected by your:

- Industry
- Business unit requirements for autonomy
- Business and accounting policies
- Business functions performed by business units and optionally, centralized in shared service centers
- Locations of facilities

Every enterprise has three fundamental structures, legal, managerial, and functional, that are used to describe its operations and provide a basis for reporting. In Oracle Fusion, these structures are implemented using the chart of accounts and organizations. Although many alternative hierarchies can be implemented and used for reporting, you are likely to have one primary structure that organizes your business into divisions, business units, and departments aligned by your strategic objectives.

Legal Structure

The figure above shows a typical group of legal entities, operating various business and functional organizations. Your ability to buy and sell, own, and employ comes from your charter in the legal system. A corporation is a distinct legal entity from its owners and managers. The corporation is owned by its shareholders, who may be individuals or other corporations. There are many other kinds of legal entities, such as sole proprietorships, partnerships, and government agencies.
A legally recognized entity can own and trade assets and employ people in the jurisdiction in which it is registered. When granted these privileges, legal entities are also assigned responsibilities to:

- Account for themselves to the public through statutory and external reporting
- Comply with legislation and regulations
- Pay income and transaction taxes
- Process value added tax (VAT) collection on behalf of the taxing authority

Many large enterprises isolate risk and optimize taxes by incorporating subsidiaries. They create legal entities to facilitate legal compliance, segregate operations, optimize taxes, complete contractual relationships, and isolate risk. Enterprises use legal entities to establish their enterprise’s identity under the laws of each country in which their enterprise operates.

In the figure above, a separate card represents a series of registered companies. Each company, including the public holding company, InFusion America, must be registered in the countries where they do business. Each company consists of various divisions created for purposes of management reporting. These are shown as vertical columns on each card. For example, a group might have a separate company for each business in the United States (US), but have their United Kingdom (UK) legal entity represent all businesses in that country. The divisions are linked across the cards so that a business can appear on some or all of the cards. For example, the air quality monitoring systems business might be operated by the US, UK, and France companies. The list of business divisions is on the Business Axis. Each company’s card is also horizontally striped by functional groups, such as the sales team and the finance team. This functional list is called the Functional Axis. The overall image suggests that information might, at a minimum, be tracked by company, business, division, and function in a group environment. In Oracle Fusion Applications, the legal structure is implemented using legal entities.

**Management Structure**

Successfully managing multiple businesses requires that you segregate them by their strategic objectives, and measure their results. Although related to your legal structure, the business organizational hierarchies do not need to be reflected directly in the legal structure of the enterprise. The management structure can include divisions, subdivisions, lines of business, strategic business units, and cost centers. In the figure above, the management structure is shown on the Business Axis. In Oracle Fusion Applications, the management structure is implemented using divisions and business units.

**Functional Structure**

Straddling the legal and business organizations is a functional organization structured around people and their competencies. For example, sales, manufacturing, and service teams are functional organizations. This functional structure is represented by the Functional Axis in the figure above. You reflect the efforts and expenses of your functional organizations directly on the income statement. Organizations must manage and report revenues, cost of sales, and functional expenses such as research and development (R&D) and selling, general, and administrative (SG&A) expenses. In Oracle Fusion Applications, the functional structure is implemented using departments and organizations, including sales, marketing, project, cost, and inventory organizations.
Enterprise Structures Business Process Model: Explained

In Oracle Fusion Applications, the Enterprise Performance and Planning Business Process Model illustrates the major implementation tasks that you perform to create your enterprise structures. This process model includes the Set Up Enterprise Structures business process, which consist of implementation activities that span many product families. Information Technology is a second Business Process Model which contains the Set Up Information Technology Management business process. Define Reference Data Sharing is one of the activities in this business process and is important in the implementation of the enterprise structures. This activity creates the mechanism to share reference data sets across multiple ledgers, business units, and warehouses, reducing the administrative burden and decreasing the time needed to implement.

The following figure and chart describes the Business Process Model structures and activities.
<table>
<thead>
<tr>
<th>BPM Activities</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define Enterprise</td>
<td>Define the enterprise to capture the name of the deploying enterprise and the location of the headquarters. There is normally a single enterprise organization in a production environment. Multiple enterprises are defined when the system is used to administer multiple customer companies, or when you choose to set up additional enterprises for testing or development.</td>
</tr>
<tr>
<td>Define Enterprise Structures</td>
<td>Define enterprise structures to represent an organization with one or more legal entities under common control. Define internal and external organizations to represent each area of business within the enterprise.</td>
</tr>
<tr>
<td>Define Legal Jurisdictions and Authorities</td>
<td>Define information for governing bodies that operate within a jurisdiction.</td>
</tr>
<tr>
<td>Define Legal Entities</td>
<td>Define legal entities and legal reporting units for business activities handled by the Oracle Fusion Applications.</td>
</tr>
<tr>
<td>Define Business Units</td>
<td>Define business units of an enterprise to allow for flexible implementation, to provide a consistent entity for controlling and reporting on transactions, and to be an anchor for the sharing of sets of reference data across applications.</td>
</tr>
<tr>
<td>Define Financial Reporting Structures</td>
<td>Define financial reporting structures, including organization structures, charts of accounts, organizational hierarchies, calendars, currencies and rates, ledgers, and document sequences which are used in organizing the financial data of a company.</td>
</tr>
<tr>
<td>Define Chart of Accounts</td>
<td>Define chart of accounts including hierarchies and values to enable tracking of financial transactions and reporting at legal entity, cost center, account, and other segment levels.</td>
</tr>
<tr>
<td>Define Ledgers</td>
<td>Define the primary accounting ledger and any secondary ledgers that provide an alternative accounting representation of the financial data.</td>
</tr>
<tr>
<td>Define Accounting Configurations</td>
<td>Define the accounting configuration that serves as a framework for how financial records are maintained for an organization.</td>
</tr>
<tr>
<td>Define Facilities</td>
<td>Define inventory, item, and cost organizations. Inventory organizations represent facilities that manufacture or store items. The item master organization holds a single definition of items that can be shared across many inventory organizations. Cost organizations group inventory organizations within a legal entity to establish the cost accounting policies.</td>
</tr>
<tr>
<td>Define Reference Data Sharing</td>
<td>Define how reference data in the applications is partitioned and shared.</td>
</tr>
</tbody>
</table>

**Note**

There are product specific implementation activities that are not listed here and depend on the applications you are implementing. For example, you can
Global Enterprise Configuration: Points to Consider

Start your global enterprise structure configuration by discussing what your organization's reporting needs are and how to represent those needs in the Oracle Fusion Applications. Consider deployment on a single instance, or at least, on as few instances as possible, to simplify reporting and consolidations for your global enterprises. The following are some questions and points to consider as you design your global enterprise structure in Oracle Fusion.

- **Enterprise Configuration**

- **Business Unit Management**

- **Security Structure**

- **Compliance Requirements**

**Enterprise Configuration**

What is the level of configuration needed to achieve the reporting and accounting requirements? What components of your enterprise do you need to report on separately? Which components can be represented by building a hierarchy of values to provide reporting at both detail and summary levels? Where are you on the spectrum of centralization versus decentralization?

**Business Unit Management**

What reporting do I need by business unit? How can you set up your departments or business unit accounts to achieve departmental hierarchies that report accurately on your lines of business? What reporting do you need to support the managers of your business units, and the executives who measure them? How often are business unit results aggregated? What level of reporting detail is required across business units?

**Security Structure**

What level of security and access is allowed? Are business unit managers and the people that report to them secured to transactions within their own business unit? Are the transactions for their business unit largely performed by a corporate department or shared service center?

**Compliance Requirements**

How do you comply with your corporate external reporting requirements and local statutory reporting requirements? Do you tend to prefer a corporate first or an autonomous local approach? Where are you on a spectrum of centralization, very centralized or decentralized?
Modeling Your Enterprise Management Structure in Oracle Fusion: Example

This example uses a fictitious global company to demonstrate the analysis that can occur during the enterprise structure configuration planning process.

Scenario

Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion enterprise resource planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers. You are chairing a committee to discuss creation of a model for your global enterprise structure including both your US and UK operations.

InFusion Corporation

InFusion Corporation has 400 plus employees and revenue of $120 million. Your product line includes all the components to build and maintain air quality monitoring (AQM) systems for homes and businesses. You have two distribution centers and three warehouses that share a common item master in the US and UK. Your financial services organization provides funding to your customers for the start up costs of these systems.

Analysis

The following are elements you need to consider in creating your model for your global enterprise structure.

- Your company is required to report using US GAAP standards and UK Statements of Standard Accounting Practice and Financial Reporting Standards. How many ledgers do you need to achieve proper statutory reporting?
- Your managers need reports that show profit and loss (revenue and expenses) for their lines of business. Do you use business units and balancing segments to represent your divisions and businesses? Do you secure data by two segments in your chart of accounts which represents each department and legal entity or one segment that represents both to produce useful, but confidential management reports?
- Your corporate management requires reports showing total organizational performance with drill down capability to the supporting details. Do you need multiple balancing segment hierarchies to achieve proper rollup of balances for reporting requirements?
- Your company has all administrative, account payables, procurement, and human resources functions performed at their corporate headquarters. Do you need one or more business unit in which to perform all these functions? How will your shared service center be configured?

The following figure and table summarize the model that your committee has designed and uses numerical values to provide a sample representation of your structure. The model includes the following recommendations:

- Creation of three separate ledgers representing your separate legal entities:
- InFusion America Inc.
- InFusion Financial Services Inc.
- InFusion UK Services Inc.
- Consolidation of results for system components, installations, and maintenance product lines across the enterprise
- All UK general and administrative costs processed at the UK headquarters
- US Systems' general and administrative costs processed at US Corporate headquarters
- US Financial Services maintains its own payables and receivables departments
In this chart, the green globe stands for mandatory and gold globe stands for optional setup. The following statements expand on the data in the chart.

- The enterprise is mandatory because it serves as an umbrella for the entire implementation. All organizations are created within an enterprise.

- Legal entities are also mandatory. They can be optionally mapped to balancing segment values or represented by ledgers. Mapping balancing segment values to legal entities is mandatory if you plan to use the intercompany functionality.

- At least one ledger is mandatory in an implementation in which you record your accounting transactions.

- Business units are also mandatory because financial transactions are processed in business units.

- A shared service center is optional, but if used, must be a business unit.

- Divisions are optional and can be represented with a hierarchy of cost centers or by a second balancing segment value.

- Departments are mandatory because they track your employees.

- Optionally, add an item master organization and inventory organizations if you are tracking your inventory transactions in Oracle Fusion Applications.

Note
Some Oracle Fusion Human Capital Management (HCM) and Customer Relationship Management (CRM) implementations do not require recording of accounting transactions and therefore, do not require implementation of a ledger.
Note

The InFusion Corporation is a legal entity but is not discussed in this example.

Define Initial Configuration with the Enterprise Structures Configurator

Establishing Enterprise Structures Using the Enterprise Structures Configurator: Explained

The Enterprise Structures Configurator is an interview-based tool that guides you through the process of setting up a basic enterprise structure. By answering questions about your enterprise, the tool creates a structure of divisions, legal entities, business units, and reference data sets that reflects your enterprise structure. After you create your enterprise structure, you also follow a guided process to determine whether or not to use positions, and whether to set up additional attributes for jobs and positions. After you define your enterprise structure and your job and position structures, you can review them, make any necessary changes, and then load the final configuration.

This figure illustrates the process to configure your enterprise using the Enterprise Structures Configurator.
To be able to use the Enterprise Structures Configurator, you must select the Enterprise Structures Guided Flow feature within the offerings in the Functional Setup Manager (FSM). If you do not select this feature, then you must set up your enterprise structure using individual tasks provided elsewhere in the offerings, and you cannot create multiple configurations to compare different scenarios.

**Establish Enterprise Structures**

To define your enterprise structures, you use the guided flow within the Establish Enterprise Structures task to enter basic information about your enterprise, such as the primary industry and the location of your headquarters. You then create divisions, legal entities, business units, and reference data sets. The Establish Enterprise Structures task enables you to create multiple enterprise configurations so that you can compare different scenarios. Until you load a configuration, you can continue to create and edit multiple configurations until you arrive at one that best suits your enterprise.

**Establish Job and Position Structures**

In addition to using the Establish Enterprise Structures task to create the basic structure of your enterprise, you also use a guided process to determine whether you want to use jobs only, or jobs and positions. The primary industry that you select in the Establish Enterprise Structures task provides the application with the information needed to make an initial recommendation. You can either accept the recommendation, or you can answer additional questions about how you manage people in your enterprise, and then make a selection. After you select whether to use jobs or positions, the guided process prompts you to set up a descriptive flexfield structure for jobs, and for positions if you have chosen to use them. Descriptive flexfields enable you to capture additional information when you create jobs and positions.

**Review Configuration**

After you create enterprise configurations and job and position structures, you can review a summary of the results of the two interview processes. For each configuration, the online summary lists the divisions, legal entities, business units, reference data sets, and job and position structures that the application will create when you load the configuration.

For a more detailed analysis of a configuration, you can access the Technical Summary Report. This report lists the same information as the online summary, but also lists the following information that will be created by the application when you load the configuration, based on your configuration:

- Legislative data groups (the application creates one legislative data group for each country that is identified in the configuration.)

- Name of the legislative data group that will be assigned to the payroll statutory unit that is generated for each legal entity.

- Organization hierarchy.

The Technical Summary report also lists the default settings that will be loaded for these fields, which you access from the Manage Enterprise HCM Information task: Worker Number Generation, Employment Model and Allow...
Employment Terms Override. You can print the Technical Summary Report for each of your configurations and compare each scenario.

Load Configuration

You can load only one configuration. When you load a configuration, the application creates the divisions, legal entities, business units, and so on. After you load the configuration, you then use individual tasks to edit, add, and delete enterprise structures.

Designing an Enterprise Configuration: Example

This example illustrates how to set up an enterprise based on a global company operating mainly in the US and the UK with a single primary industry.

Scenario

InFusion Corporation is a multinational enterprise in the high technology industry with product lines that include all the components that are required to build and maintain air quality monitoring (AQM) systems for homes and businesses. Its primary locations are in the US and the UK, but it has smaller outlets in France, Saudi Arabia, and the United Arab Emirates (UAE).

Enterprise Details

In the US, InFusion employs 400 people and has a company revenue of $120 million. Outside the US, InFusion employs 200 people and has revenue of $60 million.

Analysis

InFusion requires three divisions. The US division will cover the US locations. The Europe division will cover the UK and France. Saudi Arabia and the UAE will be covered by the Middle East division.

InFusion requires legal entities with legal employers, payroll statutory units, tax reporting units, and legislative data groups for the US, UK, France, Saudi Arabia, and UAE, in order to employ and pay its workers in those countries.

InFusion requires a number of departments across the enterprise for each area of business, such as sales and marketing, and a number of cost centers to track and report on the costs of those departments.

InFusion requires business units for human capital management (HCM) purposes. Infusion has general managers responsible for business units within each country. Those business units may share reference data. Some reference data can be defined within a reference data set that multiple business units may subscribe to. Business units are also required for financial purposes. Financial transactions are always processed within a business unit.

Resulting Enterprise Configuration

Based on this analysis, InFusion requires an enterprise with multiple divisions, ledgers, legal employers, payroll statutory units, tax reporting units, legislative data groups, departments, cost centers, and business units.
Division: Explained

Managing multiple businesses requires that you segregate them by their strategic objectives and measure their results. Responsibility to reach objectives can be delegated along the management structure. Although related to your legal structure, the business organizational hierarchies do not need to reflect directly the legal structure of the enterprise. The management entities and structure can include divisions and subdivisions, lines of business, and other strategic business units, and include their own revenue and cost centers. These organizations can be included in many alternative hierarchies and used for reporting, as long as they have representation in the chart of accounts.
Divisions

A division refers to a business oriented subdivision within an enterprise, in which each division organizes itself differently to deliver products and services or address different markets. A division can operate in one or more countries, and can be comprised of many companies or parts of different companies that are represented by business units.

A division is a profit center or grouping of profit and cost centers, where the division manager is responsible for attaining business goals including profit goals. A division can be responsible for a share of the company’s existing product lines or for a separate business. Managers of divisions may also have return on investment goals requiring tracking of the assets and liabilities of the division. The division manager reports to a top corporate executive.

By definition a division can be represented in the chart of accounts. Companies may choose to represent product lines, brands, or geographies as their divisions: their choice represents the primary organizing principle of the enterprise. This may coincide with the management segment used in segment reporting.

Oracle Fusion Applications supports a qualified management segment and recommends that you use this segment to represent your hierarchy of business units and divisions. If managers of divisions have return on investment goals, make the management segment a balancing segment. Oracle Fusion applications allows up to three balancing segments. The values of the management segment can be comprised of business units that roll up in a hierarchy to report by division.

Historically, divisions were implemented as a node in a hierarchy of segment values. For example, Oracle E-Business Suite has only one balancing segment, and often the division and legal entity are combined into a single segment where each value stands for both division and legal entity.

Use of Divisions in Oracle Fusion Human Capital Management (HCM)

Divisions are used in HCM to define the management organization hierarchy, using the generic organization hierarchy. This hierarchy can be used to create organization based security profiles.

Legal Entities: Explained

A legal entity is a recognized party with rights and responsibilities given by legislation.

Legal entities have the right to own property, the right to trade, the responsibility to repay debt, and the responsibility to account for themselves to regulators, taxation authorities, and owners according to rules specified in the relevant legislation. Their rights and responsibilities may be enforced through the judicial system. Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, and income, pay transaction taxes, or perform intercompany trading.

A legal entity has responsibility for elements of your enterprise for the following reasons:

- Facilitating local compliance
- Taking advantage of lower corporation taxation in some jurisdictions
• Preparing for acquisitions or disposals of parts of the enterprise
• Isolating one area of the business from risks in another area. For example, your enterprise develops property and also leases properties. You could operate the property development business as a separate legal entity to limit risk to your leasing business.

The Role of Your Legal Entities

In configuring your enterprise structure in Oracle Fusion Applications, you need to understand that the contracting party on any transaction is always the legal entity. Individual legal entities own the assets of the enterprise, record sales and pay taxes on those sales, make purchases and incur expenses, and perform other transactions.

Legal entities must comply with the regulations of jurisdictions, in which they register. Europe now allows for companies to register in one member country and do business in all member countries, and the US allows for companies to register in one state and do business in all states. To support local reporting requirements, legal reporting units are created and registered.

You are required to publish specific and periodic disclosures of your legal entities’ operations based on different jurisdictions’ requirements. Certain annual or more frequent accounting reports are referred to as statutory or external reporting. These reports must be filed with specified national and regulatory authorities. For example, in the United States (US), your publicly owned entities (corporations) are required to file quarterly and annual reports, as well as other periodic reports, with the Securities and Exchange Commission (SEC), who enforces statutory reporting requirements for public corporations.

Individual entities privately held or held by public companies do not have to file separately. In other countries, your individual entities do have to file in their own name, as well as at the public group level. Disclosure requirements are diverse. For example, your local entities may have to file locally to comply with local regulations in a local currency, as well as being included in your enterprise’s reporting requirements in different currency.

A legal entity can represent all or part of your enterprise’s management framework. For example, if you operate in a large country such as the United Kingdom or Germany, you might incorporate each division in the country as a separate legal entity. In a smaller country, for example Austria, you might use a single legal entity to host all of your business operations across divisions.

Creating Legal Entities in the Enterprise Structures Configurator: Points to Consider

Using the Enterprise Structures Configurator (ESC), you can create legal entities for your enterprise automatically, based on the countries in which divisions of your business operate, or you can upload a list of legal entities from a spreadsheet.

Automatically Creating Legal Entities

If you are not certain of the number of legal entities that you need, you can create them automatically. To use this option, you first identify all of the countries in
which your enterprise operates. The application opens the Map Divisions by Country page, which contains a matrix of the countries that you identified, your enterprise, and the divisions that you created. You select the check boxes where your enterprise and divisions intersect with the countries to identify the legal entities that you want the application to create. The enterprise is included for situations where your enterprise operates in a country and acts on behalf of several divisions within the enterprise and is a legal employer in a country. If you select the enterprise for a country, the application creates a country holding company.

The application automatically creates the legal entities that you select, and identifies them as payroll statutory units and legal employers. For each country that you indicated that your enterprise operates in, and for each country that you created a location for, the application also automatically creates a legislative data group.

Any legal entities that you create automatically cannot be deleted from the Create Legal Entities page within the Enterprise Structures Configurator. You must return to the Map Divisions by Country page and deselect the legal entities that you no longer want.

**Example: Creating Legal Entities Automatically**

InFusion Corporation is using the ESC to set up their enterprise structure. They have identified two divisions, one for Lighting, and one for Security. The Lighting division operates in Japan and the US, and the Security division operates in the UK and India.

This figure illustrates InFusion Corporation’s enterprise structure.

<table>
<thead>
<tr>
<th>Country</th>
<th>Enterprise</th>
<th>InFusion Lighting</th>
<th>InFusion Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Japan</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>US</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>UK</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>India</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Based on the selections made in the preceding table, the ESC creates the following four legal entities:

- InFusion Lighting Japan LE
- InFusion Lighting US LE
- InFusion Security UK LE
- InFusion Security India LE

**Creating Legal Entities Using a Spreadsheet**

If you have a list of legal entities already defined for your enterprise, you can upload them from a spreadsheet. To use this option, you first download a spreadsheet template, then add your legal entity information to the spreadsheet, and then upload directly to your enterprise configuration. You can export and import the spreadsheet multiple times to accommodate revisions.

**Legal Entity in Oracle Fusion: Points to Consider**

Oracle Fusion Applications support the modeling of your legal entities. If you make purchases from or sell to other legal entities, define these other legal entities in your customer and supplier registers, which are part of the Oracle Fusion Trading Community Architecture. When your legal entities are trading with each other, you represent both of them as legal entities and also as customers and suppliers in your customer and supplier registers. Use legal entity relationships to determine which transactions are intercompany and require intercompany accounting. Your legal entities can be identified as legal employers and therefore, are available for use in Human Capital Management (HCM) applications.

There are several decisions that need to be considered in creating your legal entities.

- The importance of legal entity in transactions
- Legal entity and its relationship to business units
- Legal entity and its relationship to divisions
- Legal entity and its relationship to ledgers
- Legal entity and its relationship to balancing segments
- Legal entity and its relationship to consolidation rules
- Legal entity and its relationship to intercompany transactions
- Legal entity and its relationship to worker assignments and legal employer
- Legal entity and payroll reporting
- Legal reporting units

**The Importance of Legal Entity in Transactions**

All of the assets of the enterprise are owned by individual legal entities. Oracle Fusion Financials allow your users to enter legal entities on transactions that represent a movement in value or obligation.
For example, the creation of a sales order creates an obligation for the legal entity that books the order to deliver the goods on the acknowledged date, and an obligation of the purchaser to receive and pay for those goods. Under contract law in most countries, damages can be sought for both actual losses, putting the injured party in the same state as if they had not entered into the contract, and what is called loss of bargain, or the profit that would have made on a transaction.

In another example, if you revalued your inventory in a warehouse to account for raw material price increases, the revaluation and revaluation reserves must be reflected in your legal entity's accounts. In Oracle Fusion Applications, your inventory within an inventory organization is managed by a single business unit and belongs to one legal entity.

**Legal Entity and Its Relationship to Business Units**

A business unit can process transactions on behalf of many legal entities. Frequently, a business unit is part of a single legal entity. In most cases the legal entity is explicit on your transactions. For example, a payables invoice has an explicit legal entity field. Your accounts payables department can process supplier invoices on behalf of one or many business units.

In some cases, your legal entity is inferred from your business unit that is processing the transaction. For example, your business unit A agrees on terms for the transfer of inventory to your business unit B. This transaction is binding on your default legal entities assigned to each business unit. Oracle Fusion Procurement, Oracle Fusion Projects, and Oracle Fusion Supply Chain applications rely on deriving the legal entity information from the business unit.

**Legal Entity and Its Relationship to Divisions**

The division is an area of management responsibility that can correspond to a collection of legal entities. If desired, you can aggregate the results for your divisions by legal entity or by combining parts of other legal entities. Define date-effective hierarchies for your cost center or legal entity segment in your chart of accounts to facilitate the aggregation and reporting by division. Divisions and legal entities are independent concepts.

**Legal Entity and Its Relationship to Ledgers**

One of your major responsibilities is to file financial statements for your legal entities. Map legal entities to specific ledgers using the Oracle Fusion General Ledger Accounting Configuration Manager. Within a ledger, you can optionally map a legal entity to one or more balancing segment values.

**Legal Entity and Its Relationship to Balancing Segments**

Oracle Fusion General Ledger supports up to three balancing segments, Best practices recommend that one of these segments represents your legal entity to ease your requirement to account for your operations to regulatory agencies, tax authorities, and investors. Accounting for your operations means you must produce a balanced trial balance sheet by legal entity. If you account for many legal entities in a single ledger, you must:
1. Identify the legal entities within the ledger

2. Balance transactions that cross legal entity boundaries through intercompany transactions

3. Decide which balancing segments correspond to each legal entity and assign them in Oracle Fusion General Ledger Accounting Configuration Manager. Once you assign one balancing segment value in a ledger, then all your balancing segment values must be assigned. This recommended best practice facilitates reporting on assets, liabilities, and income by legal entity.

Represent your legal entities by at least one balancing segment value. You may represent it by two or three balancing segment values if more granular reporting is required. For example, if your legal entity operates in multiple jurisdictions in Europe, you might define balancing segment values and map them to legal reporting units. You can represent a legal entity by more than one balancing segment value, do not use a single balancing segment value to represent more than one legal entity.

In Oracle Fusion General Ledger, there are three balancing segments. You can use separate balancing segments to represent your divisions or strategic business units to enable management reporting at the balance sheet level for each division or business unit. For example, use this solution to empower your business unit and divisional managers to track and assume responsibility for their asset utilization or return on investment. Using multiple balancing segments is also useful when you know at the time of implementation that you are disposing of a part of a legal entity and need to isolate the assets and liabilities for that entity.

**Note**

Implementing multiple balancing segments requires every journal entry that is not balanced by division or business unit, to generate balancing lines. Also, you cannot change to multiple balancing segments easily after you have begun to use the ledger because your historical data is not balanced by the new multiple balancing segments. Restating historical data must be done at that point.

To use this feature for disposal of a part of a legal entity, implement multiple balancing segments at the beginning of the legal entity’s corporate life or on conversion to Oracle Fusion.

If you decided to account for each legal entity in a separate ledger, there is no requirement to identify the legal entity with a balancing segment value within the ledger.

**Note**

While transactions that cross balancing segments don’t necessarily cross legal entity boundaries, all transactions that cross legal entity boundaries must cross balancing segments. If you make an acquisition or are preparing to dispose of a portion of your enterprise, you may want to account for that part of the enterprise in its own balancing segment even if it is not a separate legal entity. If you do not map legal entities sharing the same ledger to balancing segments, you will not be able to distinguish them using the intercompany functionality or track their individual equity.
Legal Entity and Its Relationship to Consolidation Rules

In Oracle Fusion Applications you can map legal entities to balancing segments and then define consolidation rules using your balancing segments. You are creating a relationship between the definition of your legal entities and their role in your consolidation.

Legal Entity and its Relationship to Intercompany Transactions

Use Oracle Fusion Intercompany functionality for automatic creation of intercompany entries across your balancing segments. Intercompany processing updates legal ownership within the enterprise’s groups of legal entities. Invoices or journals are created as needed. To limit the number of trading pairs for your enterprise, set up intercompany organizations and assign them to your authorized legal entities. Define processing options and intercompany accounts to use when creating intercompany transactions and to assist in consolidation elimination entries. These accounts are derived and automatically entered on your intercompany transactions based on legal entities assigned to your intercompany organizations.

Intracompany trading, in which legal ownership isn’t changed but other organizational responsibilities are, is also supported. For example, you can track assets and liabilities that move between your departments within your legal entities by creating departmental level intercompany organizations.

Note

In the Oracle Fusion Supply Chain applications, model intercompany relationships using business units, from which legal entities are inferred.

Legal Entity and Its Relationship to Worker Assignments and Legal Employer

Legal entities that employ people are called legal employers in the Oracle Fusion Legal Entity Configurator. You must enter legal employers on worker assignments in Oracle Fusion HCM.

Legal Entity and Payroll Reporting

Your legal entities are required to pay payroll tax and social insurance such as social security on your payroll. In Oracle Fusion Applications, you can register payroll statutory units to pay and report on payroll tax and social insurance on behalf of many of your legal entities. As the legal employer, you might be required to pay payroll tax, not only at the national level, but also at the local level. You meet this obligation by establishing your legal entity as a place of work within the jurisdiction of a local authority. Set up legal reporting units to represent the part of your enterprise with a specific legal reporting obligation. You can also mark these legal reporting units as tax reporting units if the legal entity must pay taxes as a result of establishing a place of business within the jurisdiction.

Business Units: Explained

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. A business unit can
process transactions on behalf of many legal entities. Normally, it will have a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss. Roll business units up into divisions if you structure your chart of accounts with this type of hierarchy. In Oracle Fusion Applications, you assign your business units to one primary ledger. For example, if a business unit is processing payables invoices they will need to post to a particular ledger. This assignment is mandatory for your business units with business functions that produce financial transactions.

In Oracle Fusion Applications, use business unit as a securing mechanism for transactions. For example, if you run your export business separately from your domestic sales business, secure the export business data to prevent access by the domestic sales employees. To accomplish this security, set up the export business and domestic sales business as two separate business units.

The Oracle Fusion Applications business unit model:

- Allows for flexible implementation
- Provides a consistent entity for controlling and reporting on transactions
- Anchors the sharing of sets of reference data across applications

Business units process transactions using reference data sets that reflect your business rules and policies and can differ from country to country. With Oracle Fusion Application functionality, you can choose to share reference data, such as payment terms and transaction types, across business units, or you can choose to have each business unit manage its own set depending on the level at which you wish to enforce common policies.

In countries where gapless and chronological sequencing of documents is required for subledger transactions, define your business units in alignment with your ledger definition, because the uniqueness of sequencing is only ensured within a ledger. In these cases, define a single ledger and assign one legal entity and business unit.

In summary, use business units in the following ways:

- Management reporting
- Processing of transactions
- Security of transactional data
- Reference data definition and sharing

**Brief Overview of Business Unit Security**

Business units are used by a number of Oracle Fusion Applications to implement data security. You assign data roles to your users to give them access to data in business units and permit them to perform specific functions on this data. When a business function is enabled for a business unit, the application can trigger the creation of data roles for this business unit base on the business function’s related job roles.

For example, if a payables invoicing business function is enabled, then it is clear that there are employees in this business unit that perform the function of payables invoicing, and need access to the payables invoicing functionality.
Therefore, based on the correspondence between the business function and the job roles, appropriate data roles are generated automatically. Use Human Capital Management (HCM) security profiles to administer security for employees in business units.

Creating Business Units in the Enterprise Structures Configurator: Points to Consider

Business units are used within Oracle Fusion applications for management reporting, processing of transactions, and security of transactional data. Using the Enterprise Structures Configurator (ESC), you create business units for your enterprise either automatically or manually.

Automatically Creating Business Units

To create business units automatically, you must specify the level at which to create business units. Business units within your enterprise may be represented at the business function level, such as Sales, Consulting, Product Development, and so on, or they may be represented at a more detailed level, where a business unit exists for each combination of countries in which you operate and the functions in those countries.

You can automatically create business units at the following levels:

- Country
- Country and Division
- Country and business function
- Division
- Division and legal entity
- Division and business function
- Business function
- Legal entity
- Business function and legal entity

Select the option that best meets your business requirements, but consider the following:

- If you use Oracle Fusion Financials, the legal entity option is recommended because of the manner in which financial transactions are processed.
- The business unit level that you select determines how the application automatically creates reference data sets.

After you select a business unit level, the application generates a list of business units, and you select the ones you want the application to create. If you select a level that has two components, such as country and division, then the system displays a table listing both components, and you select the check boxes at the intersections of the components.
The business units listed by the application are suggestions only, and are meant to simplify the process to create business units. You are not required to select all of the business units suggested. When you navigate to the next page in the ESC guided flow, which is the Manage Business Units page, you cannot delete any of the business units that were created automatically. You must return to the Create Business Units page and deselect any business units that you no longer want.

**Example: Selecting Business Unit Levels**

InFusion Corporation is using the Enterprise Structures Configurator to set up their enterprise structure. They have identified two divisions, one for Lighting, and one for Security. They operate in four countries: US, UK, Japan, and India, and they have created a legal entity for each of the countries. The sales and marketing functions are based in both India and Japan, while the US and the UK have only the sales function.

This figure illustrates InFusion Corporation’s enterprise structure.

The following table lists the options for business unit levels and the resulting business units that the application suggests for InFusion Corporation.

<table>
<thead>
<tr>
<th>Business Unit Level</th>
<th>Suggested Business Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>• US</td>
</tr>
<tr>
<td></td>
<td>• UK</td>
</tr>
<tr>
<td></td>
<td>• Japan</td>
</tr>
<tr>
<td></td>
<td>• India</td>
</tr>
<tr>
<td>Country and Division</td>
<td>• InFusion Lighting: Japan</td>
</tr>
<tr>
<td></td>
<td>• InFusion Lighting: US</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: UK</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: India</td>
</tr>
</tbody>
</table>
| Country and business function | • Sales: Japan  
• Marketing: Japan  
• Sales: US  
• Sales: UK  
• Marketing: India  
• Sales: India |
|-------------------------------|----------------|
| Division                      | • InFusion Lighting  
• InFusion Security |
| Division and Legal Entity     | • InFusion Lighting: Japan  
• InFusion Lighting: US  
• Infusion Security: UK  
• Infusion Security: India |
| Division and Business Function| • InFusion Lighting, Sales  
• InFusion Lighting, Marketing  
• InFusion Security, Sales  
• InFusion Security, Marketing |
| Business Function             | • Sales  
• Marketing |
| Legal Entity                  | • Legal Entity: Japan  
• Legal Entity: US  
• Legal Entity: UK  
• Legal Entity India |
| Legal Entity and Business Function | • Legal Entity: Japan, Sales  
• Legal Entity: Japan, Marketing  
• Legal Entity: US, Sales  
• Legal Entity: UK, Sales  
• Legal Entity India, Marketing  
• Legal Entity India, Sales |

**Manually Creating Business Units**

If none of the levels for creating business units meets your business needs, you can create business units manually, and you create them on the Manage Business Units page. If you create business units manually, then no reference data sets are created automatically. You must create them manually as well.

**Reference Data Sets and Sharing Methods: Explained**

Oracle Fusion Applications reference data sharing feature is also known as SetID. The reference data sharing functionality supports operations in multiple
ledgers, business units, and warehouses, thereby reducing the administrative burden and decreasing the time needed to implement new business units. For example, you can share sales methods, transaction types, or payment terms across business units or selected other data across asset books, cost organizations, or project units.

The reference data sharing features use reference data sets to which reference data is assigned. The reference data sets group assigned reference data. The sets can be understood as buckets of reference data assigned to multiple business units or other application components.

**Reference Data Sets**

You begin this part of your implementation by creating and assigning reference data to sets. Make changes carefully as changes to a particular set will affect all business units or application components using that set. You can assign a separate set to each business unit for the type of object that is being shared. For example, assign separate sets for payment terms, transaction types, and sales methods to your business units.

Your enterprise can decide that some aspects of corporate policy should affect all business units and leave other aspects to the discretion of the business unit manager. This allows your enterprise to balance autonomy and control for each business unit. For example, if your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level, you can let managers define their own sales methods, but define payment terms centrally. In this case, each business unit would have its own reference data set for sales methods, and there would be one central reference data set for payment terms assigned to all business units.

The reference data sharing is especially valuable for lowering the cost of setting up new business units. For example, your enterprise operates in the hospitality industry. You are adding a new business unit to track your new spa services. The hospitality divisional reference data set can be assigned to the new business unit to quickly setup data for this entity component. You can establish other business unit reference data in a business unit specific reference data set as needed.

**Reference Data Sharing Methods**

There are variations in the methods used to share data in reference data sets across different types of objects. The following list identifies the methods:

- **Assignment to one set only, no common values allowed.** The simplest form of sharing reference data that allows assigning a reference data object instance to one and only one set. For example, Asset Prorate Conventions are defined and assigned to only one reference data set. This set can be shared across multiple asset books, but all the values are contained only in this one set.

- **Assignment to one set only, with common values.** The most commonly used method of sharing reference data that allows defining reference data object instance across all sets. For example, Receivables Transaction Types are assigned to a common set that is available to all the business units without the need to be explicitly assigned the transaction types to each business unit. In addition, you can assign a business unit specific set of transaction types. At transaction entry, the list of values for transaction...
types includes transaction types from the set assigned to the business unit, as well as transaction types assigned to the common set that is shared across all business units.

- Assignment to multiple sets, no common values allowed. The method of sharing reference data that allows a reference data object instance to be assigned to multiple sets. For instance, Payables Payment Terms use this method. It means that each payment term can be assigned to one or more than one set. For example, you assign the payment term Net 30 to several sets, but the payment term Net 15 is assigned to only your corporate business unit specific set. At transaction entry, the list of values for payment terms consists of only one set of data; the set that is assigned to the transaction’s business unit.

Note: Oracle Fusion Applications contains a reference data set called Enterprise. Define any reference data that affects your entire enterprise in this set.

**Business Units and Reference Data Sets: How They Work Together**

Reference data sharing is a feature within Oracle Fusion that enables you to group set-enabled reference data such as jobs or grades so that the data can be shared across different parts of the organization. Sets also enable you to filter reference data at the transaction level so that only data that has been assigned to certain sets is available to select. To filter reference data, Oracle Fusion Human Capital Management (HCM) applications use the business unit on the transaction. To set up reference data sharing in Oracle Fusion HCM, you create business units and sets, and then assign the sets to the business units.

**Common Set Versus Specific Sets**

Some reference data in your organization may be considered global, and should therefore be made available for use within the entire enterprise. You can assign this type of data to the Common Set, which is a predefined set. Regardless of the business unit on a transaction, reference data that has been assigned to the Common Set will always be available, in addition to the reference data that has been assigned to the set that corresponds to the business unit on the transaction.

Other types of reference data may be specific to certain business units, so you want to restrict the use of the data to those business units. In this case, you can create sets specifically for this type of data, and assign the sets to the business units.

**Business Unit Set Assignment**

When you assign reference data sets to business units, you assign a default reference data set that will be used for all reference data types for that business unit. You can override the set assignment for one or more data types.

**Example: Assigning Sets to Business Units**

InFusion Corporation has two divisions: Lighting and Security, and the divisions each have two locations. Each location has one or more business functions.

The following figure illustrates the structure of InFusion Corporation.
When deciding how to create business units, InFusion decides to create them using the country and business function level. Therefore, they created the following business units:

- Sales_Japan
- Marketing_Japan
- Sales_US
- Sales_UK
- Marketing_India
- Sales_India

Because locations, departments, and grades are specific to each business unit, InFusion does not want to share these types of reference data across business units. They will create a reference data set for each business unit so that data of those types can be set up separately. Because the jobs in the Sales business function are the same across many locations, InFusion decides to create one additional set called Jobs and they will override the set assignment for the Jobs reference data group and assign it to the Jobs set. Based on these requirements, they create the following sets:

- Sales_Japan_Set
- Mktg_Japan_Set
- Sales_US_Set
- Sales_UK_Set
- Mktg_India_Set
- Sales_India_Set
- Grades_Set

InFusion assigns business units to sets as follows:

<table>
<thead>
<tr>
<th>Business Unit</th>
<th>Default Set Assignment</th>
<th>Set Assignment Overrides</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales_Japan</td>
<td>Sales_Japan_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
</tbody>
</table>
When setting up grades, departments, and locations for the business units, InFusion will assign the data to the default set for each business unit. When setting up jobs, they will assign the Jobs set and will assign the Common Set to any jobs that may be used throughout the entire organization.

When using grades, departments, and locations at the transaction level, users will be able to select data from the set that corresponds to the business unit that they enter on the transaction, and any data that was assigned to the Common Set. For example, for transactions for the Marketing_Japan business unit, grades, locations, and departments from the Mktg_Japan_Set will be available to select, as well as from the Common Set.

When using jobs at the transaction level, users will be able to select jobs from the Jobs set and from the Common Set when they enter one of the Sales business units on the transaction. For example, when a manager hires an employee for the Sales_India business unit, the list of jobs will be filtered to show jobs from the Jobs set and from the Common Set.

The following figure illustrates what sets of jobs can be accessed when a manager creates an assignment for a worker.

Creating Reference Data Sets in the Enterprise Structures Configurator:
Explained

If you created business units automatically, then the Enterprise Structures Configurator automatically creates reference data sets for you. The Enterprise
Structures Configurator creates one reference data set for each business unit. You can add additional sets, but you cannot delete any of the sets that were created automatically.

A standard set called the Enterprise set is predefined.

**Common Set**

The common set is a predefined set that enables you to share reference data across business units. When you select set-enabled data at the transaction level, the list of values includes data in both the common set and the set associated with the data type for the business unit on the transaction. For example, when you create an assignment, the list of values for grades will include both grades in the common set and in the set that is assigned to grades for the business unit in which you creating the assignment.

**Jobs and Positions: Critical Choices**

Jobs and positions represent roles that enable you to distinguish between tasks and the individuals who perform those tasks. The key to whether to use jobs or positions is how each is used. Positions offer a well-defined space independent of the person performing the job. Jobs are a space defined by the person. A job can be defined globally in the Common Set, whereas a position is defined within one business unit.

You can update the job and department of a position at any time. This is useful if you hire someone into a new role and want to transfer the position to another department.

During implementation, one of the earliest decisions you will make is whether to use jobs or a combination of jobs and positions. The determinants for this decision are:

- The primary industry of your enterprise
- How you manage your people

**Primary Industry of Your Enterprise**

Primary industries and how they usually set up their workforce are listed in the table below.

<table>
<thead>
<tr>
<th>Primary Industry</th>
<th>Workforce Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mining</td>
<td>Positions</td>
</tr>
<tr>
<td>Utilities</td>
<td>Positions</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>Positions</td>
</tr>
<tr>
<td>Retail Trade</td>
<td>Positions</td>
</tr>
<tr>
<td>Transportation and Warehousing</td>
<td>Positions</td>
</tr>
<tr>
<td>Educational Services</td>
<td>Positions</td>
</tr>
<tr>
<td>Public Transportation</td>
<td>Positions</td>
</tr>
<tr>
<td>Agriculture, Forestry, Fishing, and Hunting</td>
<td>Jobs</td>
</tr>
<tr>
<td>Construction</td>
<td>Jobs</td>
</tr>
<tr>
<td>Wholesale Trade</td>
<td>Jobs</td>
</tr>
<tr>
<td>Information</td>
<td>Jobs</td>
</tr>
<tr>
<td>Industry</td>
<td>Jobs</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Finance and Insurance</td>
<td>Jobs</td>
</tr>
<tr>
<td>Professional, Scientific, and Technical Services</td>
<td>Jobs</td>
</tr>
<tr>
<td>Management of Companies and Enterprises</td>
<td>Jobs</td>
</tr>
<tr>
<td>Administrative and Support and Waste Management and Remediation Services</td>
<td>Jobs</td>
</tr>
<tr>
<td>Arts, Entertainment, and Recreation</td>
<td>Jobs</td>
</tr>
<tr>
<td>Accommodation and Food Services</td>
<td>Jobs</td>
</tr>
<tr>
<td>Other Services (Except Public Administration)</td>
<td>Jobs</td>
</tr>
</tbody>
</table>

**Management of People**

The following table displays suggestions of whether to use jobs or a combination of jobs and positions based on your industry and how you manage your employees when there is turnover.

<table>
<thead>
<tr>
<th>Industry</th>
<th>We always replace employees by rehiring to same role</th>
<th>We replace the head count, but the manager can use the head count in a different job</th>
<th>We rehire to the same position, but the manager can request a reallocation of budget to a different post</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project (An industry that supports project-based forms of organization in which teams of specialists from both inside and outside the company report to project managers.)</td>
<td>Positions</td>
<td>Jobs</td>
<td>Jobs</td>
</tr>
<tr>
<td>Controlled (An industry that is highly structured in which all aspects of work and remuneration are well organized and regulated.)</td>
<td>Positions</td>
<td>Positions</td>
<td>Positions</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>Positions</td>
<td>Jobs</td>
<td>Positions</td>
</tr>
<tr>
<td>Retail</td>
<td>Positions</td>
<td>Jobs</td>
<td>Positions</td>
</tr>
<tr>
<td>Education</td>
<td>Positions</td>
<td>Jobs</td>
<td>Positions</td>
</tr>
<tr>
<td>Other</td>
<td>Positions</td>
<td>Jobs</td>
<td>Jobs</td>
</tr>
</tbody>
</table>

**Positions: Examples**

Positions are typically used by industries that use detailed approval rules, which perform detailed budgeting and maintain head counts, or have high turnover rates.

**Retail Industry**

ABC Corporation has high turnover. It loses approximately 5% of their cashiers monthly. The job of cashier includes three positions: front line cashier, service desk cashier, and layaway cashier. Each job is cross trained to take over another cashier position. When one cashier leaves from any of the positions, another
existing cashier from the front line, service desk or layaway can assist where needed. But to ensure short lines and customer satisfaction, ABC must replace each cashier lost to turnover.

Since turnover is high in retail it is better for this industry to use positions. There is an automatic vacancy when an employee terminates employment. The position exists even when there are no holders. This is important if the person who leaves the company is a manager or supervisor with direct reports. All direct reports continue reporting to the position even if it is empty. You do not need to reassign these employees to another manager or supervisor; the replacement manager is assigned to the existing position.

Also, an advantage to using positions is that when you hire somebody new many of the attributes are defaulted in from the position. This speeds up the hiring process.

This figure illustrates the retail position setup.

---

**Health Care Industry**

The hospital has a structured head count and detailed budgeting. For example, a specific number of surgeons, nurses, and interns of various types are needed. These positions need to be filled in order for the hospital to run smoothly. Use jobs and positions if you need to apply detailed head count rules.

Health care is an industry that needs to regulate employment, roles, and compensation according to strict policies and procedures. Fixed roles tend to endure over time, surviving multiple incumbents. Industries that manage roles rather than individuals, where roles continue to exist after individuals leave, typically model the workforce using positions.

This figure illustrates the hospital position setup.
Jobs: Example

Jobs are typically used without positions by service industries where flexibility and organizational change are key features.

Software Industry

For example, XYZ Corporation has a director over the departments for developers, quality assurance, and technical writers. Recently, three developers have left the company. The director decides to redirect the head count to other areas. Instead of hiring all three back into development, one person is hired to each department, quality assurance, and technical writing.

In software industries, the organization is fluid. Using jobs gives an enterprise the flexibility to determine where to use head count, because the job only exists through the person performing it. In this example, when the three developers leave XYZ Corporation, their jobs no longer exist, therefore the corporation has the flexibility to move the headcount to other areas.

This figure illustrates the software industry job setup.
Job and Position Structures: Explained

Job and position structures identify the descriptive flexfield structure that enables you to specify additional attributes that you want to capture when you define jobs and positions. Job and position attributes provide further detail to make jobs and positions more specific. You also use attributes to define the structure of your jobs and positions. You can specify attributes at the enterprise level for jobs and positions, at the business unit level for positions, and at the reference data set level for jobs. Job and position structures are optional.

Enterprise-Level Job Attributes

When you define a job, you enter a value for the name of the job. To make job names more specific, set up attributes that enable you to identify additional details about the job, such as the nature of the work that is performed or the relative skill level required for the job. If these attributes apply to all jobs within your enterprise, set up enterprise-level job attributes. Standard capabilities mean that you can use the different segments of the name to identify common jobs or job holders for analysis or compensation, or for grouping records in reports, for example, to find all jobs of a specific job type. You should not use attributes with values that change regularly, for example, salary ranges or expense approval levels that change every year.

This figure illustrates how job type and job level provide further details for the HR Application Specialist job.

Enterprise-Level Position Attributes

Position attributes at the enterprise level are similar to those for jobs. Each position that you define identifies a specific role in the enterprise, which you can manage independently of the person in the position, and it will belong to one specific department or organization. The name of each position must be unique. To simplify the process of managing unique names for positions, set up enterprise-level attributes to identify separate components of the position name. For example, you can set up an attribute for position title and one for position number. When defining the attributes that make up the structure of a position name you should also consider if any of your attributes are part of the definition.
of a common job type. Using job types for a position can help you manage common information that applies to many different positions. For example you can define a job type of Manager.Level 1 and use this for comparison of positions across departments or lines or business, or for setting common job requirements. You can then define multiple manager type positions in your HR department, each of which has responsibility for a different management function or group. This figure illustrates how title and position number provide further details for the manager position.

**Business Unit-Level Attributes for Positions**

If you have information that you want to capture for positions that is specific to each business unit, then you can define attributes at the business unit level for positions. When you create positions, these attributes appear in addition to any enterprise-level attributes. For example, you may want to identify the sales region for all positions in the sales business unit. You can set up a text attribute called Sales Region and use it to enter the necessary information when creating positions for the sales business unit.

**Reference Data Set-Level Attributes for Jobs**

If you have information for jobs that applies to specific reference data sets, set up attributes for jobs at the reference data set level. When you create jobs, these attributes appear in addition to any enterprise-level attributes. For example, you may want to identify all information technology (IT) jobs within a specific set. You can set up a text attribute called Function and use it to enter IT in jobs that you create that perform an IT function within a specific set.

**FAQs for Define Initial Configuration**

**What happens if I don't use the Enterprise Structures Configurator to set up my enterprise structures?**

The Enterprise Structures Configurator is an interview-based tool that guides you through setting up divisions, legal entities, business units, and reference data sets. The tool also enables you to assign reference data sets to business units and locations. You can set up multiple configurations to perform what-if
Define Common Applications Configuration for Marketing

2-43

scenarios, and then print each configuration to compare the resulting enterprise structure. If you do not use the Enterprise Structures Configurator, then you must set up your enterprise structure using the individual tasks that correspond to each enterprise component. In addition, you will not be able to set up multiple configurations and compare different scenarios. It is recommended that you use the Enterprise Structures Configurator.

What’s an ultimate holding company?

The legal entity that represents the top level in your organization hierarchy, as defined by the legal name entered for the enterprise. This designation is used only to create an organization tree, with the ultimate holding company as the top level, divisions and country holding companies as the second level, and legal employers as the third level.

What’s the default reference data set?

The reference data set that is assigned to a business unit for all reference data groups, such as grades, locations, departments, and jobs. You can override the default reference data set for any reference data group.

Define Enterprise for Marketing

Enterprise: Explained

An enterprise consists of legal entities under common control and management.

Enterprise Defined

When implementing Oracle Fusion Applications you operate within the context of an enterprise that has already been created in the application for you. This is either a predefined enterprise or an enterprise that has been created in the application by a system administrator.

An enterprise organization captures the name of the deploying enterprise and the location of the headquarters. There is normally a single enterprise organization in a production environment. Multiple enterprises are defined when the system is used to administer multiple customer companies, for example, multiple tenants, or when a customer chooses to set up additional enterprises for testing or development.

Oracle Fusion Applications offers capabilities for multiple tenants to share the same applications instance for some human resources processes. If you offer business process outsourcing services to a set of clients, each of those clients may be represented as an enterprise within an Oracle Fusion Application instance. To support this functionality, system owned reference data such as sequences, sets, and flexfields are also defined within an enterprise.

In Oracle Fusion Applications, an organization classified as an enterprise is defined before defining any other organizations in the HCM Common Organization Model. All other organizations are defined as belonging to an enterprise.

Managing Enterprise Information for Non-Oracle Fusion HCM Users: Explained

The Manage Enterprise HCM Information task includes default settings for your enterprise such as the employment model, worker number generation, and so
on. If you are not implementing Oracle Fusion Human Capital Management (HCM), then the only action you may need to perform using this task is to change the enterprise name, if necessary. The other settings are HCM-specific and are not relevant outside of Oracle Fusion HCM.

**Locations: Explained**

A location identifies physical addresses of a workforce structure, such as a department or a job. You can also create locations to enter the addresses of external organizations that you want to maintain, such as employment agencies, tax authorities, and insurance or benefits carriers.

The locations that you create exist as separate structures that you can use for reporting purposes, and also in rules that determine employee eligibility for various types of compensation and benefits. You enter information about a location only once. Subsequently, when you set up other workforce structures you select the location from a list.

**Location Sets**

When you create a location, you must associate it with a set. Only those users who have access to the set’s business unit can access the location set and other associated workforce structure sets, such as those that contain departments and jobs.

You can also associate the location to the common set so that users across your enterprise can access the location irrespective of their business unit. When users search for locations, they can see the locations that they have access to along with the locations in the common set.

The following figure shows how locations sets restrict access to users.
FAQs for Define Enterprise for Marketing

Why can't I see my location in the search results?

You can search for approved locations only. Also, if you created a location in Oracle Fusion Trading Community Model, then you can't access that location from Oracle Fusion Global Human Resources. For use in Oracle Fusion HCM, you must recreate the location from the Manage Locations page.

What happens if I select a geographic hierarchy node when I'm creating or editing a location?

The calendar events that were created for the geographical node start to apply for the location and may impact the availability of worker assignments at that location. The geographical hierarchy nodes available for selection on the Locations page display from a predefined geographic hierarchy.

What happens if I select an inventory organization when I'm creating or editing a location?

The location is available for selection in purchase documents of that inventory organization in Oracle Fusion Inventory Management. If you don't select an inventory organization, then the location is available in purchase documents across all inventory organizations.

What happens if I inactivate a location?

Starting from the effective date that you entered, you can no longer associate the location with other workforce structures, assignments, or applications. If the location is already in use, it will continue to be available to the components that currently use it.

How can I associate a location with an inventory organization?

From the Manage Locations page in Oracle Fusion Global Human Resources.

To appear on the Create or Edit Location pages, your inventory organization must be effective on today's date and must exist in the location set that you selected.

Define Geographies

Defining Address Cleansing: Explained

Address cleansing provides a way to validate, correct, and standardize addresses that are entered in a user interface. Geography validation only validates the geography attributes of an address, for example, State, City, and Postal codes; address cleansing validates both the geography attributes and the address line attributes.
Address cleansing can only be used through the Oracle Fusion Trading Community Data Quality product, because the feature is delivered using Data Quality integration. You need to ensure that you have a license for the countries that will use Trading Community Data Quality data cleansing.

You can specify the real time address cleansing level for each country by choosing either None, meaning that there is no real time address cleansing, or by choosing Optional, meaning that you will have the choice to cleanse addresses. Once you have enabled address cleansing for a country a Verify Address icon appears at address entry points in the application. You can then click the icon to perform address cleansing and receive a corrected, standardized address. If Trading Community Data Quality does not find a matching address the application will alert you.

**Geography Structure, Hierarchy, and Validation: How They Fit Together**

There are three components that are dependent on each other when defining a country: geography structure, geography hierarchy, and geography validation. Every country has to have the geography structure defined first before the hierarchy can be defined, and the geography hierarchy has to be defined before the validation can be defined.

**Geography Structure**

Firstly, you need to create a geography structure for each country to define which geography types are part of the country structure, and how the geography types are hierarchically related within the country structure. For example, you can create geography types called State, City, and Postal Code. Then you can rank the State geography type as the highest level within the country, the City as the second level, and the Postal Code as the lowest level within the country structure. Geography structure can be defined using the Manage Geographies task, or can be imported using tasks in the Define Geographies activity.

**Geography Hierarchy**

Once the geography structure is defined, the geographies for each geography type can be added to the hierarchy. For example, below the United States you can create a geography called California using a State geography type.

As part of managing the geography hierarchy you can view, create, edit, and delete the geographies for each geography type in the country structure. You can also add a primary and alternate name and code for each geography. A geography hierarchy can be created using the Manage Geographies task, or can be imported using tasks in the Define Geographies activity.

**Geography Validation**

After defining the geography hierarchy, you need to specify the geography validations for the country. You can choose which address style formats you would like to use for the country, and for each selected address style format you can map geography types to address attributes. You can also select which geography types should be included in geography or tax validation, and which geography types will display in a list of values during address entry in other user interfaces. The geography validation level for the country, such as error or warning, can also be selected.
Geography Structures: Explained

A geography structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States is the geography type of State at the top, then followed by the County, then the City, and finally the Postal Code.

You can use the geography structure to establish:

- How geographies can be related
- The types of geographies you can define for the country

How Geographies Can Be Related

You can determine how a country’s geographies are hierarchically related by creating the hierarchy of the geography types in the geography structure. When you define a country’s structure the country geography type is implicitly at the top of the geography structure, and the numbering of the subsequent levels start with 1 as the next geography level after country.

You must add a geography type as a level in the country structure before you can define a geography for that geography type in a country. For example, before defining the state of California, the State geography type must be added to the United States country structure. Only one geography type can be used for each level, you cannot define more than one geography type at the same level.

Note

After you first define a country structure you can only add geography types below the current lowest level, and delete geography types without defined geographies.

To simplify the creation of a country structure you can copy a structure from another country, and then amend the geography type hierarchy for the country.

The Types of Geographies You Can Define for the Country

The application provides you with a set of available master reference geography types. If required, you can create a geography type before adding it to the country structure. Each geography type is added below the current lowest level.

Note

If you want to delete a geography type that is not at the lowest level in the country structure, then you have to delete the geography type level and all the levels below it.

A geography type that you create within the country structure can be used for other country structures as well.

Geography Hierarchy: Explained

Geography hierarchy is a data model that lets you establish conceptual parent-child relationships between geographies. A geography, such as Tokyo or Peru,
describes a boundary on the surface of the earth. The application can extrapolate information based on this network of hierarchical geographical relationships.

For example, in the geography hierarchy the state of California is defined as the parent of San Mateo county, which is the parent of Redwood City, which is the parent of the postal code 94065. If you enter just 94065, the application can determine that the postal code is in California, or that the corresponding city is Redwood City.

The application leverages geography hierarchy information to facilitate business processes that rely on geography information, for example, tax calculation, order sourcing rules, sales territory definition. The geography hierarchy information is centrally located in the Trading Community Model and shared among other application offerings.

The top level of the geography hierarchy is Country, so the hierarchy essentially contains countries and their child geographies. Other aspects of the geography hierarchy include:

- Geography
- Geography type
- Geography usage
- Master reference geography hierarchy
- User defined zones

**Geography**

A geography is a boundary such as a country, state, province or city. It is a physical space with boundaries that is a defined instance of a geography type. For example, San Jose is a geography of the City geography type.

**Geography Type**

Geography types are a divisional grouping of geographies, which can be either geopolitical (for example, City, Province, and District) or user defined (for example, Continent, Country Regions, Tax Regions).

**Geography Usage**

Geography usage indicates how a geography type or geography is used in the application. A master reference geography always has the usage of Master Reference. User defined zones can have the usages of Tax, Shipping, or Territory, based on what is relevant for their purpose.

**Master Reference Geography Hierarchy**

The geography hierarchy data is considered to be the single source of truth for geographies. It is all the data, including geography types and geographies, that you define and maintain in the Trading Community Model tables.

The geography usage for the entire hierarchy is the master reference, and defined geography types and geographies are considered as master reference geography
types and geographies. For example, Country is a universally recognized geography type, and United States is considered a master geography.

**User Defined Zones**

User defined zones are a collection of geographical data, created from master reference data for a specific purpose. For example, territory zones are collections of master reference geographies ordered in a hierarchy. Tax and shipping zones are collections of master reference geographies without a hierarchical grouping.

**Geography Validation: Explained**

Geography validation determines the geography mapping and validation for a country's address styles, as well as the overall geography validation control for a country.

The **No Styles Format** address style format is the default address style format for a country. By defining the mapping and validation for this format you will ensure that validations can be performed for any address in the country. After the **No Styles Format** is defined you can set up additional mapping for specific address styles.

For each address style format, you can define the following:

- Map to attribute
- Enable list of values
- Tax validation
- Geography validation
- Geography validation control

**Map to Attribute**

For every address style format, you can map each geography type to an address attribute. For example, you can map the **State** geography type to the **State** address attribute for the United States, or map the **State** geography type to the **County** address attribute for the United Kingdom. The geography types that appear are based on how the country structure is defined. The list of address attributes that appear are based on address formats delivered with the application, or your customer defined address formats.

---

**Note**

You only need to map geography types that you want to use for geography or tax validation purposes.

---

**Enable List of Values**

Once a geography type is mapped to an attribute, then you can specify whether the geography type will appear in a list of values during address entry in user interfaces. It is very important to review carefully if you want to enable a list of values. You should only enable a list of values if you have sufficient geography
data imported or created for that geography. Once you have enabled a list of values for an address attribute, you can only select the geography data available for the geography type. This means that if a specific geography value is not available in the geography hierarchy, you cannot create an address with a different geography value.

**Tax Validation**

You can also specify whether a geography type will be included in tax validation. For example, for the United States North America address style format you specify that County, State, and City are used for tax validation. This will mean that when a transaction involves an address with the North America address style, the address must have the correct county, state, and city combination based on the geography hierarchy data, to be considered valid for tax calculation.

**Geography Validation**

You can specify whether a geography type will be included in geography validation. This will mean that, for example, when the user enters a United States address using the North America address style format, the address must have the correct country, state, and postal code combination based on geography hierarchy data to be considered geographically valid.

If an address element is mapped to a geography type, but not selected for geography validation usage, then suggested values can be provided for that address element during address entry, but that element is not validated.

**Note**

For either the tax or geography validation, do not skip more than one consecutive level unless you are certain that the selected geography types can uniquely identify geographies. For example, the United States country structure is: State, County, City, and Postal Code, and you want to select just State and Postal Code for geography or tax validation. However, for the combination of California and 94065, the city can be either Redwood Shores or Redwood City. In this case, you should also select at least the City geography type for geography or tax validation.

**Geography Validation Control**

You can select the geography validation level for a country. Validation will check if the entered address maps to the geography hierarchy data available for the country, and the geography validation control determines whether you can save an address that did not pass validation during address entry. For example, if the validation level is **Warning**, then an address can still be saved if the values do not match the geography hierarchy data.

These are the geography validation levels you can choose:

- **Error** - only completely valid addresses can be saved, with all mandatory address elements entered.
- **No Validation** - all addresses can be saved including incomplete and invalid addresses.
Warning - invalid addresses are saved after warning users.

Regardless of the result of validation, the validation process will try to map any address attribute to a geography of the country, and store any mapping it could establish based on the available data. This is called Geography Name Referencing and it is executed as part of validation. The result of this referencing is used in several business processes in the application to map an address to a specific geography or zone.

Zone Types and Zones: Explained

You can create zone types and zones for the use of defining boundaries to be used in, for example, tax or shipping zones.

In order to create a zone boundary you need to define the following:

- Zone types
- Zones

Zone Types

Zone types categorize and group zones together, for example, the zone types of Income Tax and Shipping Regions.

Zone types need to be created before you define a zone for the geographical boundary. You can create a zone type which will contain geographical boundaries from anywhere in the world, or you can create a zone type that will only contain geographies from within a specified country. When you create a zone type that is bounded by a country you can define which geography types or geographies you will be able to choose when you create a zone.

Zones

After you have created the zone type click Next and you will be able to add zones. Zones are geographical boundaries for a zone type, for example, the San Jose Tax zone. Zones are based on the master reference geography hierarchy data.

Zones are created within a zone type, and you can associate geographies to define the zone. For example, for the Shipping Regions zone type you can create a West Coast zone which has the state of California as one of its geographies. Within a geography you can specify a postal range. So for the state of California, for example, you can specify that the zone spans from postal code 90001 to 90011.

Zone Hierarchy: Explained

Using the territory manager zone hierarchy you can build a zone hierarchy by creating zones and zone types, and by adding master reference geographies. The zones, master geographies, and hierarchies can then be used, for example, by Territory Management to define a sales region or geographical boundary that is allocated to a salesperson.

In a zone hierarchy you can do the following:

- Create zone types
• Create zones and add to a hierarchy
• Move zones or geographies
• Add geographies to a hierarchy

Create Zone Types

When you are creating a zone you need to specify a zone type. Zone types categorize and group zones together, for example, an APAC zone type. You will need to choose if the zone is part of an existing zone type, or if not, then you will need to create a new zone type. After you have created or added a zone type to the zone you can enter the zone name, the zone code name, and the zone’s effective dates.

Create Zones and Add to a Hierarchy

You can create zones to describe geographical boundaries, for example, the Singapore Sales zone and the Southwest Sales Region zone. Zones can be placed below another zone or geography in the hierarchy, and geographies can be placed below a zone.

Move Zones or Geographies

You can move existing zones or geographies into your hierarchy. You can select the zone or geography you want the zone to appear below, and then select an existing zone that you want to move. The zone and all its child records will appear below the zone or geography you selected.

Add Geographies to a Hierarchy

You can create a hierarchy using the geographies from the master reference geography hierarchy data, and you can also add geographies to hierarchies created from zones.

When you are adding a geography to a hierarchy you have the option of either adding just the geography, or you can add the geography and selected child geographies. All the child geographies you select will automatically be added to the hierarchy, and will reflect the master reference geographical hierarchy. For example, when adding the United Kingdom geography to a hierarchy you can select that all the counties and postal codes will be added. When the hierarchy is generated the counties will be the level above the postal codes.

Note

You cannot have the same geography in more than one hierarchy.

Define Geographies: Define File-Based Data Import

Files, Import Objects, Mapping, and Import Activity Components: How They Work Together

File-based import supports the import of data from an external text or xml file to interface tables and then from interface tables to target application tables.
Overview of File-Based Data Import

File-based import includes the following:

- Source files with import data
- Import objects with available import attributes
- Mappings between source files and interface table columns
- Import Activities to define import options, a processing schedule, and monitor progress

Source Files

External data can be obtained in various ways and formatted in a text or xml file. The source file data is mapped to interface table columns using a Mapping. The source file is identified on an Import Activity, along with other import processing details. The file processing component of the file-based data import consists of reading the source file, parsing the data, and inserting the data into the appropriate interface tables.

Objects

Import objects are defined where interface tables exist and external files can be used to import data into the interface tables. Import Object definitions for Oracle objects that support file-based import are predefined and can be accessed with the appropriate security privilege. Individual object attributes represent the interface table columns and are used to map source file data or constant values in Mappings and Import Activity definitions. Use the Import Object definition to manage the display of attributes that can be mapped, to indicate required mappings, and to set site level default values as required.

Mappings

Import mapping enables you to predefine a mapping between the columns provided in a source file and the attributes pertaining to the objects being imported. Once you create a mapping, it can be reused in the Import Activity definition.

Manage Import Activities

An Import Activity definition provides the instructions for the import processing. It includes the source file or file location and mapping, plus import processing options and schedule. You can monitor the progress of the Import Activity processing and view completion reports for both successful records and errors.

File-Based Import Processing: How it Works

The file-based data import process includes processing the source file data and inserting it into the interface tables, moving the interface table data into the destination application tables, and then processing the attachments for the imported objects. Processing factors are subject to the settings defined for the Import Activity, Mapping, and Import Object. You can monitor the processing steps and view process reports for each Import Activity.
This topic describes the following:

- Inserting Data in the Interface Tables
- Interface Table Data Validation and Error Counts
- Interface Table to Destination Application Table Processing
- Importing Attachments
- Viewing Import Results

**Inserting Data in the Interface Tables**

Data exists in various sources and in various formats. The file import processing starts with reading the source data, parsing the data, and inserting into the appropriate interface tables. The source of the data comes from the following:

- Source file values mapped to target object attributes in the Import Activity.
- Constant values defined for target object attributes in the Import Activity.
- Default values defined for target object attributes in the Import Object.

**Interface Table Data Validation and Error Counts**

The data is initially validated against the predefined Import Mapping and the Import Object settings as the interface tables are being populated by the initial file import process. The interface table data is validated again before importing into the destination application tables.

- Validation includes:
  - Missing required values
  - Values that exceed the attribute length
  - Invalid values
  - Duplicates to existing records in the destination application tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping.

**Note**

For the Lead import object, the duplicate checking is only done for existing leads created within the look back days setting of the Import Activity.

- Duplicates to existing records in the destination application tables for Customer Data Management objects based on Matching Configurations.
- Errors
  
  Most validation issues are recorded as errors, with the exception of Customer Data Management duplicates found during the Matching
Configuration process. In this case, matched records are only considered as errors if:

- Customer Management Duplicates option is set to **Do Not Import** for the Import Activity and
- The main object of the Import Activity is a consumer, customer, or legal entity object
- **Allowable Error Count Threshold**

The validation of the interface table occurs before any records are imported into the destination application tables. Once the validation process has completed, the count of records with errors is compared to the Allowable Error Count Threshold value specified for the Import Activity. A count above the threshold will stop the import process for all records. If the count is below the threshold, records without errors will import. In either case, records with errors will be reported in the Error and Exception files.

**Interface Table to Destination Application Table Processing**

The import process orchestrates the import for each of the component objects that make up the overall main objects of the Import Activity.

**Importing Attachments**

Once the objects have imported successfully, the attachments are processed. The import process matches the source file attachment name to the file name included in the compressed file entered on the Import Activity. The attachment file is imported into Universal Content Manager and then associated as an attachment to the imported object.

**Viewing Import Results**

You can monitor all file-based Import Activities that are currently scheduled to run, have completed successfully, or failed with errors. For each Import Activity, you can view the details pertaining to each underlying process. Once an Import Activity process has completed, the following processing reports are added as attachments to the process:

- Log file. Includes the records that were successfully imported plus the unique destination application table identifiers for the objects.
- Exception file. Includes the records that were not imported plus a reference to an error for each record that failed validation.
- Error file. Includes all the errors for each record that failed validation.

**File-Based Import Objects: Explained**

Import objects represent the application and attribute information for business objects that can be imported using external source files.

This topic describes the following:

- Import object management options
- Custom objects

**Import Object Management Options**

A single import object can have multiple associated components that are considered objects by themselves. An object and associated objects that can be imported within the same source file are grouped together within the application module class.

**Note**

Each object includes the Import Activity object (MktImpJobs1). The Import Activity object is a required component of the application module but is not mapped to a source file. All values for this object are derived from the Import Activity definition. Consequently, do not update the **Map**, **Required**, and **Default Value** settings for the Import Activity object.

The following table includes information about the import object:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes</td>
<td>A view-only listing of object attributes that represent each column in the interface table for the object.</td>
</tr>
<tr>
<td>Length</td>
<td>A view-only listing of widths for the columns in the interface tables. If the source file values for the attribute have more characters than the attribute length, the source file row will not be imported.</td>
</tr>
<tr>
<td>Default Value</td>
<td>Optionally, specify an attribute value to use if a value is not available from the source file or Import Activity constant value.</td>
</tr>
<tr>
<td>Map</td>
<td>Enable the list of attributes that can be mapped to a source file or constant value in the Import Mapping and Import Activity Map Fields step.</td>
</tr>
<tr>
<td>Required</td>
<td>Specify the list of attributes that must be mapped to source file columns. Consequently, if you have selected an attribute as required, you must also enable the <strong>Map</strong> option for that attribute. When mapping the external source file, the required target attribute defined for the object are displayed with an asterisk.</td>
</tr>
</tbody>
</table>

**Custom Objects**

To use the file-based import feature for custom objects, you must first generate the artifacts required for import. You generate these required artifacts within Oracle Fusion CRM Application Composer, after making your object model extensions.

**File-Based Import Mapping: Explained**

Import mapping enables you to predefined a mapping between the columns provided in a source file and the attributes pertaining to the objects being
imported. Once you create a mapping, it can be reused in the Import Activity definition.

This topic contains the following sections:

- Import options
- Source file options
- Target options

**Import Options**

The following attributes pertain to the import mapping.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The business object to be imported.</td>
</tr>
<tr>
<td>Name</td>
<td>The name that identifies the mapping in the Import Mapping and Import Activity UIs. If the mapping was initially created while mapping fields directly in the Import Activity user interface and automatically saved without providing a user-defined mapping name, the mapping name is derived from the Import Activity name and date.</td>
</tr>
<tr>
<td>Decimal Separator</td>
<td>The format of the fractional portion of numerical values in columns mapped to attributes with a decimal attribute type.</td>
</tr>
<tr>
<td>Date Format</td>
<td>The format of values in columns mapped to attributes with a date attribute type.</td>
</tr>
<tr>
<td>Timestamp Format</td>
<td>The format of values in columns mapped to attributes with a time stamp attribute type.</td>
</tr>
<tr>
<td>Lock</td>
<td>If selected, prevents any user, other than the creator of the mapping, from editing the mapping.</td>
</tr>
</tbody>
</table>

**Source File Options**

Map each column that the source file is expected to contain with a specific attribute.

The following table describes the details pertaining to columns provided in the source file:

<table>
<thead>
<tr>
<th>Source Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence</td>
<td>The sequence number in which the columns are expected to be provided in the source file. Two rows cannot have the same sequence number.</td>
</tr>
<tr>
<td>Column Name</td>
<td>The column name expected in the source file if a header row is included, or more generic values such as Column A, Column B, and so on, if the header row is not included for Text file types. The tagging structure is represented for XML file types.</td>
</tr>
<tr>
<td>Column Width</td>
<td>Use when the delimiter value is fixed width for Text file types only.</td>
</tr>
</tbody>
</table>
### Ignore
- Ignore the source file column to exclude the data from being imported.

### Required
- If selected, a value must exist in the source file or the row will not be imported.

---

### Target Options
The following table describes the details pertaining to corresponding attributes in the target application table:

<table>
<thead>
<tr>
<th>Target Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The group of import objects that represent the components of the business object being imported.</td>
</tr>
<tr>
<td>Attribute</td>
<td>The attribute name that represents the corresponding interface table column for the object.</td>
</tr>
<tr>
<td>Duplicate Validation</td>
<td>If selected, the attribute, along with other selected attributes, determines what constitutes a duplicate object when comparing objects in the interface tables and existing objects in the target application tables. For example, to validate the uniqueness of an object in the target application tables by the combination of an object's name and date, select Duplicate Validation for both attributes in the mapping.</td>
</tr>
</tbody>
</table>

---

### Import Activity Source File Options: Explained

The Import Activity consists of a step by step guided process to assist you with creating an import activity for a given object.

This topic describes the source file options defined in the Import Activity that are used by the import process to locate and parse the source file data.

#### Source File Data
Enter attribute details pertaining to the source file as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Type</td>
<td>Source file must be either Text or XML.</td>
</tr>
<tr>
<td>Data Type, Delimiter, and Header Row Included</td>
<td>A Text file type can further be defined based on how the data is delimited and if the source file is expected to include a row of headings for each column.</td>
</tr>
<tr>
<td>Import Mapping</td>
<td>Displays a list of predefined mappings for the object selected for this import activity. The selected mapping will be used as the basis for mapping your source file in the next Import Activity step.</td>
</tr>
</tbody>
</table>

---

### Source File Location
The following outlines the options that are available to you when locating your source file for import.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Selection</td>
<td>Select from the following file selections:</td>
</tr>
<tr>
<td></td>
<td>• Specific file</td>
</tr>
<tr>
<td></td>
<td>Enables you to upload a specific source file from a local file system, such as your desktop, a URL address, or from a network path. A file name is required for this option.</td>
</tr>
<tr>
<td></td>
<td>• Most recent file</td>
</tr>
<tr>
<td></td>
<td>Enables you to schedule repeating import activities without having to select a new file every time. This selection is only available when you select <strong>Network</strong> from the <strong>Upload From</strong> options.</td>
</tr>
<tr>
<td></td>
<td>You need to copy the new file to the specified network path for repeating import activities. You do not need to enter a file name for this option and can only upload your source file from a network path. The asterisk wildcard is supported for multiple characters. The question mark wildcard is supported for a single character.</td>
</tr>
<tr>
<td>Upload From</td>
<td>You can upload the source file from three locations:</td>
</tr>
<tr>
<td></td>
<td>• Desktop</td>
</tr>
<tr>
<td></td>
<td>• URL</td>
</tr>
<tr>
<td></td>
<td>• Network</td>
</tr>
<tr>
<td></td>
<td>If you select <strong>Desktop</strong>, a <strong>File Name</strong> field with an associated Update button is displayed. Click <strong>Update</strong> and browse to search for and select the file you want to upload.</td>
</tr>
<tr>
<td></td>
<td>If you select <strong>URL</strong>, enter the address location as in the following example format: <code>http://www.example.com/</code></td>
</tr>
<tr>
<td></td>
<td>If you select <strong>Network</strong>, enter the file name path as in the following example format: <code>\ComputerName\SharedFolder\Resource\</code></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>If you selected the <strong>Specific File</strong> as your file selection option, then you will have to include the file name for both URL and Network file path locations.</td>
</tr>
</tbody>
</table>
In the Summary section enter an import activity name and description and the primary object for which you want to import data from a source file. The list of objects displayed is controlled by your data security privileges. This topic describes how to define the attachment options pertaining to the import activity.

Attachments

As part of the file attachment import process setup, you must:

- Provide the relationship between the attachment file or files and the object in the source file with multiple columns each referencing a file name pertaining to the attachment.
- Provide file names in the source file columns for attachment corresponding to each object record (row).
- Select all the files associated with all the objects targeted in the current file import activity process.
- Map the columns related to file names to specific object and attribute pertaining to the common attachment interface such as category, file name, file title and file description.
- Monitor the process for uploading attachments that is activated as part of the file import activity process.

Selecting Documents

You define the parameters for the import activity to include the primary object for which the data is included in the source file. If the object being imported has attachments, you will need perform an additional step of selecting documents that serve as attachments for each record being imported in the Attachments section. Select the Multiple Files option and then click on Browse to display the Universal Content Manager (UCM). From here you can select individual documents that serve as attachments or a single file that contains all these documents as follows:

- Select a pre-configured compressed file in Zip or Jar format that contains all the individual attachment documents. If the compressed file contains hierarchy of folders then the attachment import process will traverse through the hierarchy to search for specific file name.
- Select individual attachment documents which UCM automatically compresses into a Zip format. In this case, the individual document cannot be a compressed file.

Browse through the file system and select multiples files from across various folders. You must select all attachments in one operation. For example, you cannot select a few files now and then return later to select more attachments files.

Import Activity Import Options: Explained

The File Import Activity consists of a step by step guided process to assist you with creating an import activity for a given object.

This topic describes the import options defined in the Import Activity that are used by the import process to interpret source file data and import interface table data into the target application tables.

Source File Data Transformation

The following options are used to identify the formatting of source file data so the data can be correctly interpreted and transformed by the import process:
### Define Common Applications Configuration for Marketing

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decimal Separator</td>
<td>The format of the fractional portion of numerical values in columns mapped to attributes with a decimal attribute type.</td>
</tr>
<tr>
<td>Date Format</td>
<td>The format for values in columns mapped to attributes with a date attribute type.</td>
</tr>
<tr>
<td>Time Stamp Format</td>
<td>The format for values in columns mapped to attributes with a time stamp attribute type.</td>
</tr>
<tr>
<td>File Encoding</td>
<td>The overall encoding of the characters within the file.</td>
</tr>
</tbody>
</table>

### Interface to Target Import Options

The following options are used when importing the interface table information to the target application tables:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Mode</td>
<td>Determines if the Import Activity process should create new records or update existing records.</td>
</tr>
<tr>
<td></td>
<td>If updating existing records, the record IDs must be provided in the source file. If an existing record is not found, a new record is created. Update mode is not supported for all import objects. Consequently, the <strong>Import Mode</strong> is set to <strong>Create</strong> and is not updatable for those objects.</td>
</tr>
<tr>
<td></td>
<td>If creating new records, the import process evaluates the data in the interface tables with existing objects in the target application tables for possible duplicates. Customer Data Management objects are evaluated using the rules defined in the set of Matching Configurations. All other objects are evaluated using the combination of attributes selected for duplicate validation in the predefined Import Mapping.</td>
</tr>
<tr>
<td>Allowable Error Count</td>
<td>An error count above the threshold will stop the import process for all records. If the error count is below the threshold, records without errors are imported. In either case, records with errors will be reported in the Error and Exception files. Validation errors include:</td>
</tr>
<tr>
<td></td>
<td>• Missing required values</td>
</tr>
<tr>
<td></td>
<td>• Values that exceed the attribute length</td>
</tr>
<tr>
<td></td>
<td>• Invalid identifiers and lookup codes</td>
</tr>
<tr>
<td></td>
<td>• Duplicates to existing records in the destination tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping</td>
</tr>
<tr>
<td></td>
<td>Duplicates found using matching configurations for Customer Data Management objects do not contribute to the error count.</td>
</tr>
<tr>
<td><strong>Notification E-Mail</strong></td>
<td>The e-mail of the intended recipient of import processing notifications.</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td><strong>Customer Data Management Duplicates</strong></td>
<td>Consumer, customer, and legal entity objects imported by themselves or as components of another object are subject to duplicate verification. The duplicates are determined using the following matching configurations:</td>
</tr>
<tr>
<td></td>
<td>- Batch Location Basic Duplicate Identification</td>
</tr>
<tr>
<td></td>
<td>- Batch Person Basic Duplicate Identification</td>
</tr>
<tr>
<td></td>
<td>- Batch Organization Basic Duplicate Identification</td>
</tr>
<tr>
<td>You can select from one of the following:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Do Not Import Duplicate Records</strong></td>
</tr>
<tr>
<td></td>
<td>If the main object of the Import Activity is a consumer, customer, or a legal entity object, rows that are matched to existing records will not be imported. These duplicates records are reported in the Exception and Error reports.</td>
</tr>
<tr>
<td></td>
<td>If the Customer Data Management objects are components of another object and one or more matches are found, the existing duplicate records are evaluated to determine the most recent record. The most recent record will be associated with the main object being imported.</td>
</tr>
<tr>
<td></td>
<td>For example, when importing a marketing response object, the consumer object is also a component of the response. If the consumer is matched to an existing record, the consumer in the interface tables is not imported. However, the response object will import and the most recent existing consumer record will be associated to the response.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Import Duplicate Records</strong></td>
</tr>
<tr>
<td></td>
<td>The Customer Data Management objects will be imported even if matched records exist.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Import Duplicate Records and Create Resolution Request</strong></td>
</tr>
<tr>
<td></td>
<td>The Customer Data Management objects will be imported even if matched records exist. In addition, a duplicate resolution request is created and displayed in the Customer Data Management, Duplicate Resolution work area.</td>
</tr>
</tbody>
</table>
Duplicate Look Back Days

This option applies only to the Lead import object. Only existing leads created within the period determined by the look back days value are evaluated for duplicates based on the attributes selected for duplicate validation in the predefined import mapping. If a duplicate is found, the lead will not be imported and the duplicate record will be reported on the Exception report. Duplicate leads are included in the calculation of the allowable error count threshold.

Import Activity Field Mapping: Explained

After entering your import options, the second step of the import activity process is to map fields in the source file to the corresponding target attributes. This topic explains:

- Map Fields
- Saving the Import Mapping
- Constant Values

Map Fields

The Map Fields section can be subdivided into source file columns and target attribute columns.

The source column header value is derived from one of the following:

- Predefined mapping, if one is selected
- The source file, if the Header Row Included option is selected in the first step of the Import Activity definition (for Text file type only)
- Generic values of Column A, Column B, and so on, if the Header Row Included option is not selected (for Text file type only)
- XML tagging structure (for XML file type only)

The following table outlines the source columns:

<table>
<thead>
<tr>
<th>Source Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Header</td>
<td>Represents the column header for Text file types and the tagging structure for XML file types.</td>
</tr>
<tr>
<td>Example Value</td>
<td>Values are derived from the first source file saved with the predefined mapping. If you did not select a predefined mapping, the example values are taken from the first data row in the source file selected in the first step of the Import Activity definition.</td>
</tr>
<tr>
<td>Ignore</td>
<td>Select this option if you do not want to import the source file data in that column.</td>
</tr>
</tbody>
</table>

The following table outlines the target columns:

<table>
<thead>
<tr>
<th>Target Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The group of import objects that represent the components of the business object being imported.</td>
</tr>
<tr>
<td>Attribute</td>
<td>The attribute name that represents the corresponding interface table column for the object.</td>
</tr>
</tbody>
</table>
Saving the Import Mapping

The mapping between source file information and target attributes is saved as a reusable mapping when the Import Activity is saved, using the import activity name and date to derive a mapping name. If you selected a predefined mapping, modifications made in the Import Activity to an unlocked mapping will update and save to the predefined mapping. If the predefined mapping is locked, a modified mapping will be saved as a new mapping. To specify a mapping name for new mappings, select the **Save As** option from the Map Fields **Actions** menu.

Constant Values

Constant values provide a way to specify a value for a target attribute that all imported objects will inherit. For example, if a source file does not contain a column for business unit and all of the objects in the file belong to the same business unit, enter a constant value for the object and business unit attribute.

File-Based Import Monitoring: Explained

You can monitor all file import activities that are currently scheduled to run, have completed successfully, or failed with errors. For each import activity, you can view the details pertaining to each underlying process and make necessary updates for any failed records to import again.

You can view the list of import activities from the Manage Import Activities page. Select the import activity that you want to monitor by clicking on the hyperlink in the corresponding Status column. The View Import Status results page is displayed which contains the following sections:

- Files Processed
- Import Processes

Files Processed

The Files Processed section displays a row for each source file that is processed. The import processing details are summarized and displayed for each source file and include the following:

<table>
<thead>
<tr>
<th>File Processing Summary Information</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records Read From File</td>
<td>The number of records read from the source file.</td>
</tr>
<tr>
<td>Format Errors</td>
<td>The number of errors found when processing data to insert into the interface tables from the source file, Import Activity constants, and Import Object value default values. View the error details in the Exception and Error files attached to the process.</td>
</tr>
<tr>
<td>Load Errors</td>
<td>The number of errors found when importing data from the interface tables to the destination application tables. View the error details in the Exception and Error files attached to the process.</td>
</tr>
<tr>
<td>Successfully Loaded</td>
<td>The number of import objects imported to the application destination tables. If the import object is made up of multiple components, each component is counted as successfully loaded. Consequently the Successfully Loaded count may be larger than the Records Read From File count. View the successful record details in the Log file attached to the process.</td>
</tr>
</tbody>
</table>
Once an Import Activity process has completed, processing reports are included in the Attachments column. The Log file includes the records that were successfully imported plus the unique destination application table identifiers for the objects. The Exception file includes the records that were not imported plus a reference to one of the errors for each record that failed. The Error file includes all the errors for each record that failed validation.

**Import Processes**

From the Import Processes section, you can view details pertaining to each process involved in importing the objects in the source file. A listing of brief messages provides information on processing steps within each underlying process.

**Define Geographies: FAQs for File-Based Data Import**

**What determines the list of objects displayed?**

A single import object can have multiple associated components that are considered objects by themselves. Whether or not an associated object can be grouped as a component of another object for the purpose of file import is determined by the complexity of the object structure and how it is stored in the data model. Oracle Fusion provides import objects predefined to meet the file processing import requirements. Consequently, in some cases, more than one source file may be required to capture all associated components of an object.

**What happens if I inactivate an Import Activity?**

The Import Activity will not stop the currently running process. However, it will stop the next process that has not started plus any future repeating file import activities. You can always activate the process at a later stage.

**What happens if I add a marketing list in the Import Activity definition?**

File-based data import enables you to record consumers and organization contacts in a marketing list when importing consumer, lead, and response import objects. Select an existing list or create a new one. A marketing list is assigned the list type value of Imported if created while defining an import activity. After the objects are imported successfully, the consumers and contacts are added as members of the marketing list.

**Define Legal Entities for Marketing**

**Legal Entities: Explained**

A legal entity is a recognized party with rights and responsibilities given by legislation.

Legal entities have the right to own property, the right to trade, the responsibility to repay debt, and the responsibility to account for themselves to regulators, taxation authorities, and owners according to rules specified in the relevant legislation. Their rights and responsibilities may be enforced through the judicial system. Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, and income, pay transaction taxes, or perform intercompany trading.

A legal entity has responsibility for elements of your enterprise for the following reasons:
• Facilitating local compliance
• Taking advantage of lower corporation taxation in some jurisdictions
• Preparing for acquisitions or disposals of parts of the enterprise
• Isolating one area of the business from risks in another area. For example, your enterprise develops property and also leases properties. You could operate the property development business as a separate legal entity to limit risk to your leasing business.

**The Role of Your Legal Entities**

In configuring your enterprise structure in Oracle Fusion Applications, you need to understand that the contracting party on any transaction is always the legal entity. Individual legal entities own the assets of the enterprise, record sales and pay taxes on those sales, make purchases and incur expenses, and perform other transactions.

Legal entities must comply with the regulations of jurisdictions, in which they register. Europe now allows for companies to register in one member country and do business in all member countries, and the US allows for companies to register in one state and do business in all states. To support local reporting requirements, legal reporting units are created and registered.

You are required to publish specific and periodic disclosures of your legal entities’ operations based on different jurisdictions’ requirements. Certain annual or more frequent accounting reports are referred to as statutory or external reporting. These reports must be filed with specified national and regulatory authorities. For example, in the United States (US), your publicly owned entities (corporations) are required to file quarterly and annual reports, as well as other periodic reports, with the Securities and Exchange Commission (SEC), who enforces statutory reporting requirements for public corporations.

Individual entities privately held or held by public companies do not have to file separately. In other countries, your individual entities do have to file in their own name, as well as at the public group level. Disclosure requirements are diverse. For example, your local entities may have to file locally to comply with local regulations in a local currency, as well as being included in your enterprise’s reporting requirements in different currency.

A legal entity can represent all or part of your enterprise’s management framework. For example, if you operate in a large country such as the United Kingdom or Germany, you might incorporate each division in the country as a separate legal entity. In a smaller country, for example Austria, you might use a single legal entity to host all of your business operations across divisions.

**Legal Entity in Oracle Fusion: Points to Consider**

Oracle Fusion Applications support the modeling of your legal entities. If you make purchases from or sell to other legal entities, define these other legal entities in your customer and supplier registers, which are part of the Oracle Fusion Trading Community Architecture. When your legal entities are trading with each other, you represent both of them as legal entities and also as customers and suppliers in your customer and supplier registers. Use legal entity relationships to determine which transactions are intercompany and require intercompany accounting. Your legal entities can be identified as legal employers and therefore, are available for use in Human Capital Management (HCM) applications.
There are several decisions that need to be considered in creating your legal entities.

- The importance of legal entity in transactions
- Legal entity and its relationship to business units
- Legal entity and its relationship to divisions
- Legal entity and its relationship to ledgers
- Legal entity and its relationship to balancing segments
- Legal entity and its relationship to consolidation rules
- Legal entity and its relationship to intercompany transactions
- Legal entity and its relationship to worker assignments and legal employer
- Legal entity and payroll reporting
- Legal reporting units

The Importance of Legal Entity in Transactions

All of the assets of the enterprise are owned by individual legal entities. Oracle Fusion Financials allow your users to enter legal entities on transactions that represent a movement in value or obligation.

For example, the creation of a sales order creates an obligation for the legal entity that books the order to deliver the goods on the acknowledged date, and an obligation of the purchaser to receive and pay for those goods. Under contract law in most countries, damages can be sought for both actual losses, putting the injured party in the same state as if they had not entered into the contract, and what is called loss of bargain, or the profit that would have made on a transaction.

In another example, if you revalued your inventory in a warehouse to account for raw material price increases, the revaluation and revaluation reserves must be reflected in your legal entity’s accounts. In Oracle Fusion Applications, your inventory within an inventory organization is managed by a single business unit and belongs to one legal entity.

Legal Entity and Its Relationship to Business Units

A business unit can process transactions on behalf of many legal entities. Frequently, a business unit is part of a single legal entity. In most cases the legal entity is explicit on your transactions. For example, a payables invoice has an explicit legal entity field. Your accounts payables department can process supplier invoices on behalf of one or many business units.

In some cases, your legal entity is inferred from your business unit that is processing the transaction. For example, your business unit A agrees on terms for the transfer of inventory to your business unit B. This transaction is binding on your default legal entities assigned to each business unit. Oracle Fusion Procurement, Oracle Fusion Projects, and Oracle Fusion Supply Chain applications rely on deriving the legal entity information from the business unit.

Legal Entity and Its Relationship to Divisions

The division is an area of management responsibility that can correspond to a collection of legal entities. If desired, you can aggregate the results for your divisions by legal entity or by combining parts of other legal entities. Define date-effective hierarchies for your cost center or legal entity segment in your chart of accounts to facilitate the aggregation and reporting by division. Divisions and legal entities are independent concepts.
Legal Entity and Its Relationship to Ledgers

One of your major responsibilities is to file financial statements for your legal entities. Map legal entities to specific ledgers using the Oracle Fusion General Ledger Accounting Configuration Manager. Within a ledger, you can optionally map a legal entity to one or more balancing segment values.

Legal Entity and Its Relationship to Balancing Segments

Oracle Fusion General Ledger supports up to three balancing segments, Best practices recommend that one of these segments represents your legal entity to ease your requirement to account for your operations to regulatory agencies, tax authorities, and investors. Accounting for your operations means you must produce a balanced trial balance sheet by legal entity. If you account for many legal entities in a single ledger, you must:

1. Identify the legal entities within the ledger
2. Balance transactions that cross legal entity boundaries through intercompany transactions
3. Decide which balancing segments correspond to each legal entity and assign them in Oracle Fusion General Ledger Accounting Configuration Manager. Once you assign one balancing segment value in a ledger, then all your balancing segment values must be assigned. This recommended best practice facilitates reporting on assets, liabilities, and income by legal entity.

Represent your legal entities by at least one balancing segment value. You may represent it by two or three balancing segment values if more granular reporting is required. For example, if your legal entity operates in multiple jurisdictions in Europe, you might define balancing segment values and map them to legal reporting units. You can represent a legal entity by more than one balancing segment value, do not use a single balancing segment value to represent more than one legal entity.

In Oracle Fusion General Ledger, there are three balancing segments. You can use separate balancing segments to represent your divisions or strategic business units to enable management reporting at the balance sheet level for each division or business unit. For example, use this solution to empower your business unit and divisional managers to track and assume responsibility for their asset utilization or return on investment. Using multiple balancing segments is also useful when you know at the time of implementation that you are disposing of a part of a legal entity and need to isolate the assets and liabilities for that entity.

Note

Implementing multiple balancing segments requires every journal entry that is not balanced by division or business unit, to generate balancing lines. Also, you cannot change to multiple balancing segments easily after you have begun to use the ledger because your historical data is not balanced by the new multiple balancing segments. Restating historical data must be done at that point.

To use this feature for disposal of a part of a legal entity, implement multiple balancing segments at the beginning of the legal entity’s corporate life or on conversion to Oracle Fusion.

If you decided to account for each legal entity in a separate ledger, there is no requirement to identify the legal entity with a balancing segment value within the ledger.
Note
While transactions that cross balancing segments don't necessarily cross legal entity boundaries, all transactions that cross legal entity boundaries must cross balancing segments. If you make an acquisition or are preparing to dispose of a portion of your enterprise, you may want to account for that part of the enterprise in its own balancing segment even if it is not a separate legal entity. If you do not map legal entities sharing the same ledger to balancing segments, you will not be able to distinguish them using the intercompany functionality or track their individual equity.

Legal Entity and Its Relationship to Consolidation Rules
In Oracle Fusion Applications you can map legal entities to balancing segments and then define consolidation rules using your balancing segments. You are creating a relationship between the definition of your legal entities and their role in your consolidation.

Legal Entity and its Relationship to Intercompany Transactions
Use Oracle Fusion Intercompany functionality for automatic creation of intercompany entries across your balancing segments. Intercompany processing updates legal ownership within the enterprise’s groups of legal entities. Invoices or journals are created as needed. To limit the number of trading pairs for your enterprise, set up intercompany organizations and assign them to your authorized legal entities. Define processing options and intercompany accounts to use when creating intercompany transactions and to assist in consolidation elimination entries. These accounts are derived and automatically entered on your intercompany transactions based on legal entities assigned to your intercompany organizations.

Intracompany trading, in which legal ownership isn’t changed but other organizational responsibilities are, is also supported. For example, you can track assets and liabilities that move between your departments within your legal entities by creating departmental level intercompany organizations.

Note
In the Oracle Fusion Supply Chain applications, model intercompany relationships using business units, from which legal entities are inferred.

Legal Entity and Its Relationship to Worker Assignments and Legal Employer
Legal entities that employ people are called legal employers in the Oracle Fusion Legal Entity Configurator. You must enter legal employers on worker assignments in Oracle Fusion HCM.

Legal Entity and Payroll Reporting
Your legal entities are required to pay payroll tax and social insurance such as social security on your payroll. In Oracle Fusion Applications, you can register payroll statutory units to pay and report on payroll tax and social insurance on behalf of many of your legal entities. As the legal employer, you might be required to pay payroll tax, not only at the national level, but also at the local level. You meet this obligation by establishing your legal entity as a place of work within the jurisdiction of a local authority. Set up legal reporting units to represent the part of your enterprise with a specific legal reporting obligation. You can also mark these legal reporting units as tax reporting units if the legal entity must pay taxes as a result of establishing a place of business within the jurisdiction.
HCM Organization Models: Examples

These examples illustrate different models for human capital management (HCM) organizations. Each example includes a legislative data group (LDG). LDGs are not an organization classification, but they are included in the example to show how you associate them with a payroll statutory unit to partition payroll data.

Simple Configuration

This example illustrates a simple configuration that does not include any tax reporting units. The legal employer and payroll statutory units are the same, sharing the same boundaries. Reporting can only be done at a single level. Countries such as Saudi Arabia and the United Arab Emirates (UAE) might use this type of model, as reporting in these countries is done at the legal entity level.

This figure illustrates a simple configuration where the enterprise has only one legal entity that is both a payroll statutory unit and a legal employer.

Multiple Legal Employers and Tax Reporting Units Under One Payroll Statutory Unit

This example illustrates a more complex configuration. In this enterprise, one legal entity, InFusion US, is defined as a payroll statutory unit and has two separate legal entities, which are also legal employers. This model shows multiple legal employers that are associated with a single payroll statutory unit, and how tax reporting units are always associated with a specific legal employer (or employers) through the payroll statutory unit. The implication is that payroll
statutory reporting boundaries vary from human resources (HR) management, and the balances can be categorized separately by either payroll statutory unit, legal employer, or tax reporting unit. This configuration is based on tax filing requirements, as some tax-related payments and reports are associated with a higher level than employers. An example of a country that might use this model is the US.

This figure illustrates an enterprise that has one payroll statutory unit and multiple legal employers and tax reporting units.

**One Payroll Statutory Unit and Two Tax Reporting Units That Are Subsidiaries of the Legal Entity**

This model makes no distinction between a legal employer and a payroll statutory unit. Tax reporting units are defined as subsidiaries to the legal entity.
In this enterprise, legal entity is the highest level of aggregation for payroll calculations and reporting, and statutory reporting boundaries are assumed to be the same for both payroll and HR management. An example of a country that might use this model is France.

This figure illustrates an example of an organization with one legal entity that is both a legal employer and a payroll statutory unit and that has two tax reporting units.

In this model, the enterprise has one legal entity, and legal employers and tax reporting units are independent from each other within a payroll statutory unit, because there is no relationship from a legal perspective. Therefore, you can run reporting on both entities independently. Using this model, you would not typically need to report on tax reporting unit balances within a legal employer, and balances can be categorized by either or both organizations, as required. An example of a country that might use this model is India.

This figure illustrates an enterprise with one legal entity that is a payroll statutory unit and a legal employer, and the tax reporting units are independent from the legal employer.
Multiple Payroll Statutory Units with Several Tax Reporting Units that are Independent from the Legal Employer

In this model, the enterprise has two legal entities, and legal employers and tax reporting units are independent from each other within a payroll statutory unit, because there is no relationship from a legal perspective. Therefore, you can run reporting on both entities independently. Using this model, you would not typically need to report on tax reporting unit balances within a legal employer, and balances can be categorized by either or both organizations, as required. An example of a country that might use this model is the United Kingdom (UK).

This figure illustrates an enterprise with two legal entities, and legal employers and tax reporting units are independent from each other.
Payroll Statutory Units, Legal Employers, and Tax Reporting Units: How They Work Together

When you set up legal entities, you can identify them as legal employers and payroll statutory units, which makes them available for use in Oracle Fusion Human Capital Management (HCM). A tax reporting unit is created automatically when you add a legal entity and identify it as a payroll statutory unit. Depending on how your organization is structured, you may have only one legal entity that is also a payroll statutory unit and a legal employer, or you may have multiple legal entities, payroll statutory units, and legal employers.

Legal Employers and Payroll Statutory Unit

Payroll statutory units enable you to group legal employers so that you can perform statutory calculations at a higher level, such as for court orders or for
United Kingdom (UK) statutory sick pay. In some cases, a legal employer is also a payroll statutory unit. However, your organization may have several legal employers under one payroll statutory unit. A legal employer can belong to only one payroll statutory unit.

**Payroll Statutory Units and Tax Reporting Units**

Payroll statutory units and tax reporting units have a parent-child relationship, with the payroll statutory unit being the parent.

**Tax Reporting Units and Legal Employers**

Tax reporting units are indirectly associated with a legal employer through the payroll statutory unit. One or more tax reporting units can be used by a single legal employer, and a tax reporting unit can be used by one or more legal employers. For example, assume that a single tax reporting unit is linked to a payroll statutory unit. Assume also that two legal employers are associated with this payroll statutory unit. In this example, both legal employers are associated with the single tax reporting unit.

**FAQs for Define Legal Entities for Marketing**

**What's a legal employer?**

A legal employer is a legal entity that employs workers. You define a legal entity as a legal employer in the Oracle Fusion Legal Entity Configurator.

The legal employer is captured at the work relationship level, and all employment terms and assignments within that relationship are automatically with that legal employer. Legal employer information for worker assignments is also used for reporting purposes.

**What's a payroll statutory unit?**

Payroll statutory units are legal entities that are responsible for paying workers, including the payment of payroll tax and social insurance. A payroll statutory unit can pay and report on payroll tax and social insurance on behalf of one or many legal entities, depending on the structure of your enterprise. For example, if you are a multinational, multicompany enterprise, then you register a payroll statutory unit in each country where you employ and pay people. You can optionally register a consolidated payroll statutory unit to pay and report on workers across multiple legal employers within the same country. You associate a legislative data group with a payroll statutory unit to provide the correct payroll information for workers.

**Define Business Units for Marketing**

**Shared Service Centers: Explained**

Oracle Fusion Applications allows defining relationships between business units to outline which business unit provides services to the other business units.

**Service Provider Model**

In Oracle Fusion Applications V1.0, the service provider model centralizes only the procurement business function. Your business units that have the
requisitioning business function enabled can define relationships with business units that have the procurement business function enabled. These service provider business units will process requisitions and negotiate supplier terms for their client business units.

This functionality is used to frame service level agreements and drive security. The definition of service provider relationships provides you with a clear record of how the operations of your business are centralized. For other centralized processing, business unit security is used (known in Oracle EBS as Multi-Org Access Control). This means that users who work in a shared service center have the ability to get access and process transactions on behalf of many business units.

Shared Service Center: Points to Consider

Oracle Fusion applications supports shared service centers in two ways. First, with business unit security, which allows your shared service centers personnel to process transactions for other business units called clients. This was the foundation of Multi Org Access Control in the Oracle E-Business Suite.

Second, the service provider model expands on this capability to allow a business unit and its personnel in a shared service center to work on transactions of the client business units. It is possible to view the clients of a service provider business unit, and to view service providers of a client business unit.

Your shared service centers provide services to your client business units that can be part of other legal entities. In such cases, your cross charges and recoveries are in the form of receivables invoices, and not merely allocations within your general ledger, thereby providing internal controls and preventing inappropriate processing.

For example, in traditional local operations, an invoice of one business unit cannot be paid by a payment from another business unit. In contrast, in your shared service center environment, processes allowing one business unit to perform services for others, such as paying an invoice, are allowed and completed with the appropriate intercompany accounting. Shared service centers provide your users with access to the data of different business units and can comply with different local requirements.

Security

The setup of business units provides you with a powerful security construct by creating relationships between the functions your users can perform and the data they can process. This security model is appropriate in a business environment where local business units are solely responsible for managing all aspects of the finance and administration functions.

In Oracle Fusion applications, the business functions your business unit performs are evident in the user interface for setting up business units. To accommodate shared services, use business unit security to expand the relationship between functions and data. A user can have access to many business units. This is the core of your shared service architecture.

For example, you take orders in many business units each representing different registered legal entities. Your orders are segregated by business unit. However, all of these orders are managed from a shared service order desk in an
outsourcing environment by your users who have access to multiple business units.

Benefits

In summary, large, medium, and small enterprises benefit from implementing share service centers. Examples of functional areas where shared service centers are generally implemented include procurement, disbursement, collections, order management, and human resources. The advantages of deploying these shared service centers are the following:

- Reduce and consolidate the number of control points and variations in processes, mitigating the risk of error.
- Increase corporate compliance to local and international requirements, providing more efficient reporting.
- Implement standard business practices, ensuring consistency across the entire enterprise and conformity to corporate objectives.
- Establish global processes and accessibility to data, improving managerial reporting and analysis.
- Provide quick and efficient incorporation of new business units, decreasing startup costs.
- Establish the right balance of centralized and decentralized functions, improving decision making.
- Automate self-service processes, reducing administrative costs.
- Permit business units to concentrate on their core competencies, improving overall corporate profits.

Service Provider Model: Explained

In Oracle Fusion applications, the service provider model defines relationships between business units for a specific business function, identifying one business in the relationship as a service provider of the business function, and the other business unit as its client.

Procurement Example

The Oracle Fusion Procurement product family has taken advantage of the service provide model by defining outsourcing of the procurement business function. Define your business units with requisitioning and payables invoicing business functions as clients of your business unit with the procurement business function. Your business unit responsible for the procurement business function will take care of supplier negotiations, supplier site maintenance, and purchase order processing on behalf of your client business units. Subscribe your client business units to the supplier sites maintained by the service providers, using a new procurement feature for supplier site assignment.

In the InFusion example below, business unit four (BU4) serves as a service provider to the other three business units (BU1, BU2, and BU3.) BU4 provides the corporate administration, procurement, and human resources (HR) business functions, thus providing cost savings and other benefits to the entire InFusion enterprise.
Business Functions: Explained

A business unit can perform many business functions in Oracle Fusion Applications. Prior to Oracle Fusion Applications, operating units in Oracle E-Business Suite were assumed to perform all business functions, while in PeopleSoft, each business unit had one specific business function. Oracle Fusion Applications blends these two models and allows defining business units with one or many business functions.

Business Functions

A business function represents a business process, or an activity that can be performed by people working within a business unit and describes how a business unit is used. The following business functions exist in Oracle Fusion applications:

- Billing and revenue management
- Collections management
- Customer contract management
- Customer payments
- Expense management
- Incentive compensation
- Marketing
- Materials management
- Inventory management
- Order fulfillment orchestration
- Payables invoicing
• Payables payments
• Procurement
• Procurement contract management
• Project accounting
• Receiving
• Requisitioning
• Sales

Although there is no relationship implemented in Oracle Fusion Applications, a business function logically indicates a presence of a department in the business unit with people performing tasks associated with these business functions. A business unit can have many departments performing various business functions. Optionally, you can define a hierarchy of divisions, business units, and departments as a tree over HCM organization units to represent your enterprise structure.

**Note**

This hierarchy definition is not required in the setup of your applications, but is a recommended best practice.

Your enterprise procedures can require a manager of a business unit to have responsibility for their profit and loss statement. However, there will be cases where a business unit is performing only general and administrative functions, in which case your manager’s financial goals are limited to cost containment or recovering of service costs. For example, if a shared service center at the corporate office provides services for more commercially-oriented business units, it does not show a profit and therefore, only tracks its costs.

In other cases, where your managers have a responsibility for the assets of the business unit, a balance sheet can be produced. The recommended best practice to produce a balance sheet is to setup the business unit as a balancing segment in the chart of accounts. The business unit balancing segment can roll up to divisions or other entities to represent your enterprise structure.

When a business function produces financial transactions, a business unit must be assigned to a primary ledger, and a default legal entity. Each business unit can post transactions to a single primary ledger, but it can process transactions for many legal entities.

**Business Unit Hierarchy: Example**

For example, your InFusion America Company provides:

• Air quality monitoring systems through your division InFusion Air Systems
• Customer financing through your division InFusion Financial Services

The InFusion Air Systems division further segments your business into the System Components and Installation Services subdivisions. Your subdivisions are divided by business units:

• System Components by products: Air Compressors and Air Transmission
• Installation Services by services: Electrical and Mechanical
Oracle Fusion applications facilitates independent balance sheet rollups for legal and management reporting by offering up to three balancing segments. Hierarchies created using the management segment can provide the divisional results. For example, it is possible to define management segment values to correspond to business units, and arrange them in a hierarchy where the higher nodes correspond to divisions and subdivisions, as in the Infusion US Division example above.

**Business Units: Explained**

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. A business unit can process transactions on behalf of many legal entities. Normally, it will have a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss. Roll business units up into divisions if you structure your chart of accounts with this type of hierarchy. In Oracle Fusion Applications, you assign your business units to one primary ledger. For example, if a business unit is processing payables invoices they will need to post to a particular ledger. This assignment is mandatory for your business units with business functions that produce financial transactions.

In Oracle Fusion Applications, use business unit as a securing mechanism for transactions. For example, if you run your export business separately from your domestic sales business, secure the export business data to prevent access by the domestic sales employees. To accomplish this security, set up the export business and domestic sales business as two separate business units.

The Oracle Fusion Applications business unit model:

- Allows for flexible implementation
Define Common Applications Configuration for Marketing

- Provides a consistent entity for controlling and reporting on transactions
- Anchors the sharing of sets of reference data across applications

Business units process transactions using reference data sets that reflect your business rules and policies and can differ from country to country. With Oracle Fusion Application functionality, you can choose to share reference data, such as payment terms and transaction types, across business units, or you can choose to have each business unit manage its own set depending on the level at which you wish to enforce common policies.

In countries where gapless and chronological sequencing of documents is required for subledger transactions, define your business units in alignment with your ledger definition, because the uniqueness of sequencing is only ensured within a ledger. In these cases, define a single ledger and assign one legal entity and business unit.

In summary, use business units in the following ways:

- Management reporting
- Processing of transactions
- Security of transactional data
- Reference data definition and sharing

Brief Overview of Business Unit Security

Business units are used by a number of Oracle Fusion Applications to implement data security. You assign data roles to your users to give them access to data in business units and permit them to perform specific functions on this data. When a business function is enabled for a business unit, the application can trigger the creation of data roles for this business unit base on the business function’s related job roles.

For example, if a payables invoicing business function is enabled, then it is clear that there are employees in this business unit that perform the function of payables invoicing, and need access to the payables invoicing functionality. Therefore, based on the correspondence between the business function and the job roles, appropriate data roles are generated automatically. Use Human Capital Management (HCM) security profiles to administer security for employees in business units.

Define Workforce Structures for CRM

Locations: Explained

A location identifies physical addresses of a workforce structure, such as a department or a job. You can also create locations to enter the addresses of external organizations that you want to maintain, such as employment agencies, tax authorities, and insurance or benefits carriers.

The locations that you create exist as separate structures that you can use for reporting purposes, and also in rules that determine employee eligibility for various types of compensation and benefits. You enter information about a location only once. Subsequently, when you set up other workforce structures you select the location from a list.

Location Sets

When you create a location, you must associate it with a set. Only those users who have access to the set’s business unit can access the location set and other
associated workforce structure sets, such as those that contain departments and jobs.

You can also associate the location to the common set so that users across your enterprise can access the location irrespective of their business unit. When users search for locations, they can see the locations that they have access to along with the locations in the common set.

The following figure shows how locations sets restrict access to users.

---

**Division: Explained**

Managing multiple businesses requires that you segregate them by their strategic objectives and measure their results. Responsibility to reach objectives can be delegated along the management structure. Although related to your legal structure, the business organizational hierarchies do not need to reflect directly the legal structure of the enterprise. The management entities and structure can include divisions and subdivisions, lines of business, and other strategic business units, and include their own revenue and cost centers. These organizations can be included in many alternative hierarchies and used for reporting, as long as they have representation in the chart of accounts.

**Divisions**

A division refers to a business oriented subdivision within an enterprise, in which each division organizes itself differently to deliver products and services or address different markets. A division can operate in one or more countries, and can be comprised of many companies or parts of different companies that are represented by business units.

A division is a profit center or grouping of profit and cost centers, where the division manager is responsible for attaining business goals including profit goals. A division can be responsible for a share of the company’s existing...
product lines or for a separate business. Managers of divisions may also have return on investment goals requiring tracking of the assets and liabilities of the division. The division manager reports to a top corporate executive.

By definition a division can be represented in the chart of accounts. Companies may choose to represent product lines, brands, or geographies as their divisions: their choice represents the primary organizing principle of the enterprise. This may coincide with the management segment used in segment reporting.

Oracle Fusion Applications supports a qualified management segment and recommends that you use this segment to represent your hierarchy of business units and divisions. If managers of divisions have return on investment goals, make the management segment a balancing segment. Oracle Fusion applications allows up to three balancing segments. The values of the management segment can be comprised of business units that roll up in a hierarchy to report by division.

Historically, divisions were implemented as a node in a hierarchy of segment values. For example, Oracle E-Business Suite has only one balancing segment, and often the division and legal entity are combined into a single segment where each value stands for both division and legal entity.

**Use of Divisions in Oracle Fusion Human Capital Management (HCM)**

Divisions are used in HCM to define the management organization hierarchy, using the generic organization hierarchy. This hierarchy can be used to create organization based security profiles.

**Cost Centers and Departments: Explained**

A cost center represents the smallest segment of an organization for which costs are collected and reported. A department is an organization with one or more operational objectives or responsibilities that exist independently of its manager and has one or more workers assigned to it.

The following two components need to be considered in designing your enterprise structure:

- Cost centers
- Departments

**Cost Centers**

A cost center also represents the destination or function of an expense as opposed to the nature of the expense which is represented by the natural account. For example, a sales cost center indicates that the expense goes to the sales department.

A cost center is generally attached to a single legal entity. To identify the cost centers within a chart of accounts structure use one of these two methods:

- Assign a cost center value in the value set for each cost center. For example, assign cost center values of PL04 and G3J1 to your manufacturing teams in the US and India. These unique cost center values allow easy aggregation of cost centers in hierarchies (trees) even if the cost centers are in different ledgers. However, this approach will require defining more cost center values.
- Assign a balancing segment value with a standardized cost center value to create a combination of segment values to represent the cost center. For
example, assign the balancing segment values of 001 and 013 with cost center PL04 to represent your manufacturing teams in the US and India. This creates 001-PL04 and 013-PL04 as the cost center reporting values.

The cost center value of PL04 has a consistent meaning. This method requires fewer cost center values to be defined. However, it prevents construction of cost center hierarchies using trees where only cost center values are used to report results for a single legal entity. You must specify a balancing segment value in combination with the cost center values to report on a single legal entity.

Departments

A department is an organization with one or more operational objectives or responsibilities that exist independently of its manager. For example, although the manager may change, the objectives do not change. Departments have one or more workers assigned to them.

A manager of a department is typically responsible for:

- Controlling costs within their budget
- Tracking assets used by their department
- Managing employees, their assignments, and compensation

Note

The manager of a sales department may also be responsible for meeting the revenue targets.

The financial performance of departments is generally tracked through one or more cost centers. In Oracle Fusion Applications, departments are defined and classified as Department organizations. Oracle Fusion Human Capital Management (HCM) assigns workers to departments, and tracks the headcount at the departmental level.

The granularity of cost centers and their relationship to departments varies across implementations. Cost center and department configuration may be unrelated, identical, or consist of many cost centers tracking the costs of one department.

Department Classifications: Points to Consider

A department can be classified as a project organization, sales and marketing organization, or cost organization.

Oracle Fusion Human Capital Management (HCM) uses trees to model organization hierarchies. It provides seeded tree structures for department and other organizational hierarchies that can include organizations with any classification.

Project Organization

Classify departments as a project owning organization to enable associating them with projects or tasks. The project association is one of the key drivers for project access security.

In addition, you must classify departments as project expenditure organizations to enable associating them to project expenditure items. Both project owning
organizations and project expenditure organizations can be used by Oracle Fusion Subledger Accounting to derive accounts for posting Oracle Fusion Projects accounting entries to Oracle Fusion General Ledger.

**Sales and Marketing Organization**

In Oracle Fusion Customer Relationship Management (CRM), you can define sales and marketing organizations. Sales organization hierarchies are used to report and forecast sales results. Sales people are defined as resources assigned to these organizations.

In some enterprises, the HCM departments and hierarchies correspond to sales organizations and hierarchies. It is important to examine the decision on how to model sales hierarchies in relationship to department hierarchies when implementing customer relationship management to eliminate any possible redundancy in the definition of the organizations.

The following figure illustrates a management hierarchy, in which the System Components Division tracks its expenses in two cost centers, Air Compressors and Air Transmission. At the department level, two organizations with a classifications of Department are defined, the Marketing Department and Sales Department. These two departments can be also identified as a Resource Organizations, which will allow assigning resources, such as sales people, and other CRM specific information to them. Each department is represented in the chart of accounts by more than one cost center, allowing for granular as well as hierarchical reporting.

---

**Cost Organization**

Oracle Fusion Costing uses a cost organization to represent a single physical inventory facility or group of inventory storage centers, for example, inventory...
organizations. This cost organization can roll up to a manager with responsibility for the cost center in the financial reports.

A cost organization can represent a costing department. Consider this relationship when determining the setup of departments in HCM. There are no system dependencies requiring these two entities, cost organization and costing department, be set up in the same way.

**Jobs: Example**

Jobs are typically used without positions by service industries where flexibility and organizational change are key features.

**Software Industry**

For example, XYZ Corporation has a director over the departments for developers, quality assurance, and technical writers. Recently, three developers have left the company. The director decides to redirect the head count to other areas. Instead of hiring all three back into development, one person is hired to each department, quality assurance, and technical writing.

In software industries, the organization is fluid. Using jobs gives an enterprise the flexibility to determine where to use head count, because the job only exists through the person performing it. In this example, when the three developers leave XYZ Corporation, their jobs no longer exist, therefore the corporation has the flexibility to move the headcount to other areas.

This figure illustrates the software industry job setup.

**FAQs for Define Workforce Structures for CRM**

**Why can't I see my location in the search results?**

You can search for approved locations only. Also, if you created a location in Oracle Fusion Trading Community Model, then you can't access that location from Oracle Fusion Global Human Resources. For use in Oracle Fusion HCM, you must recreate the location from the Manage Locations page.
How can I associate a location with an inventory organization?

From the Manage Locations page in Oracle Fusion Global Human Resources.

To appear on the Create or Edit Location pages, your inventory organization must be effective on today’s date and must exist in the location set that you selected.

What happens if I select an inventory organization when I’m creating or editing a location?

The location is available for selection in purchase documents of that inventory organization in Oracle Fusion Inventory Management. If you don’t select an inventory organization, then the location is available in purchase documents across all inventory organizations.

What happens if I select a geographic hierarchy node when I’m creating or editing a location?

The calendar events that were created for the geographical node start to apply for the location and may impact the availability of worker assignments at that location. The geographical hierarchy nodes available for selection on the Locations page display from a predefined geographic hierarchy.

What happens if I inactivate a location?

Starting from the effective date that you entered, you can no longer associate the location with other workforce structures, assignments, or applications. If the location is already in use, it will continue to be available to the components that currently use it.

What’s the difference between a job set and a job family?

A job family is a group of jobs that have different but related functions, qualifications, and titles. They are beneficial for reporting. You can define competencies for job families by associating them with model profiles.

A job set is an organizational partition of jobs. For example, a job set can be global and include jobs for use in all business units, or it can be restricted to jobs for a specific country or line of business. When you select a job, for a position or an assignment, the available jobs are those in the set associated with the business unit in which you are working, and also those in the Common set.

Define Facilities for Marketing

How Schedule Components Fit Together

Schedules are comprised of workday patterns and exceptions. Workday patterns are comprised of shifts.
First, create shifts and then assign them to workday patterns. Next, create a schedule that is a collection of workday patterns. You can also create exceptions to assign to schedules. All of these components are illustrated in the following diagram:

**Shift**

A shift is a period of time, typically expressed in hours, and it can be defined by a start time and an end time, or a duration. A shift can be for a work period or an off period. You can create time, duration, and elapsed shifts.

**Workday Pattern**

A workday pattern is a collection of shifts for a specific number of days. You can create time, duration, and elapsed workday patterns.
Exception

An exception is a record of a date that overrides the availability of a resource to which a schedule has been assigned. For example, a resource is assigned a schedule that includes December 25 as a working day. An exception can be created for December 25 and applied to that schedule to override resource availability for that date. Exceptions can also be for a date time period such as 9 AM to 11 AM on December 25th.

Schedule

A schedule is defined by a start date, an end date, and a sequence of workday patterns to be followed between those dates. A schedule can also contain exception dates that override the availability of resources to which the schedule is assigned. Quarter Types such as 4-4-5, 4-5-4 are supported.

Managing Shifts: Examples

A shift is a period of time, typically expressed in hours, that is used to build workday patterns. Workday patterns are used to build schedules. There are multiple types of shifts you can create. The following scenarios illustrate each type:

Managing Time Shifts

Next month you are adding a second shift for your manufacturing operations. This new shift will start right after your regular first shift. You can create a time shift that starts at 4:00 p.m. and ends at 12:00 a.m. There are restrictions in updating existing shifts and patterns. Shifts and patterns cannot be updated if they effect a schedule. If a shift is created but not assigned to a pattern (or assigned to a pattern but pattern is not assigned to a schedule) it can be updated. If a pattern is created and not assigned to a schedule it can be updated.

Managing Time Shifts with Punch Details

Your division has decided that the employees in the office must clock in and out for lunch starting next week. All employees will take the same lunch hour. Add punch shift details to the existing shift so that employees punch in at 8:00 a.m.; they punch out for lunch from 11:30 a.m. to 12:30 p.m.; they punch back in at 12:30 p.m.; and they punch out for the day at 5:00 p.m.

Managing Time Shifts with Flexible Details

Jorge Sanchez is a contractor who is starting work in your department next week. His hours will be flexible, so you need to create a new time shift with flexible details that he can use to record his time. He will have a flexible start time from 7:00 a.m. to 9:00 a.m. and a flexible end time from 4:00 p.m. to 6:00 p.m. His core work hours will be from 9:00 a.m. to 4:00 p.m.

Managing Duration Shifts

One of the divisions in your organization does not use fixed start and end times for its daily shifts; the division only records the total duration of the shift and indicates if resources are available or not during that time. All of the employees in the division are available for 24 hours straight, and then they are not available for the next 24 hours. You should create a duration shift that indicates that
resources are available for 24 hours, and create a second duration shift that indicates that resources are not available for 24 hours.

**Managing Elapsed Shifts**

The employees in the Human Resources department all work 8 hours a day, but the start and end times vary by employee. Some employees start at early as 6:00 a.m.; while others don’t start until 9:00 a.m. Create an elapsed shift with a duration of 8 hours, where all employees are assumed to be available for the number of hours in the shift at any time during the day.

**Managing Workday Patterns: Examples**

A workday pattern is a collection of shifts for a specific number of days. There are multiple types of workday patterns you can create. The following scenarios illustrate each type:

**Managing Time Workday Patterns**

Your department works a Monday through Friday workweek with 8 hour shifts each day. Time patterns always have time shifts. That is, the shift will have start time and end time. You can create a time workday pattern with a length of 7 days and details of an 8 hour time shift for days 1 through 5. Days 6 and 7 are considered nonworking days. Remember that time patterns always have time shifts, that is, have a start time and an end time.

**Managing Duration Workday Patterns**

A new group of employees start next month, and each employee will work a schedule where he or she is available for 10 hours, and then not available for the next 16 hours, and then available for 10 hours again, and so on. This pattern starts on midnight of the first day of the next month. Create a duration workday pattern with a 10 hour available duration shift, followed by a 16 hour not available duration shift. Do not specify the pattern length or start and end days, and the pattern will repeat for the length of the schedule to which it is associated.

**Managing Elapsed Workday Patterns**

In the summer, several divisions in your organization work only 4 hours on Fridays. They work extended hours on Wednesdays and Thursdays to cover the 4 hours they will not work on Fridays. Create an elapsed workday pattern with a length of 7 days. Days 1 and 2 will have an 8 hour shift assigned; while days 3 and 4 will have a 10 hour shift assigned. Finally, day 5 will have a 4 hour shift assigned. As in the time workday pattern, days 6 and 7 are considered nonworking days.

**Item Organization: Explained**

An item organization defines an item when inventory balances are not stored and inventory storage or inventory movement is not reflected in the Oracle Fusion Applications. For example, you would use an item organization in a retail scenario, if you need to know the items that are listed by and sold through each retail outlet even though inventory and transactions are recorded in another system. In Oracle Fusion Customer Relationship Management (CRM), item organizations are used to define sales catalogs.
Note

- Items belong to an item organization.
- Item attributes that are associated with financial and accounting information are hidden from the item if it exists within the item organization.
- Item organizations can be changed by administrators to an inventory organization by updating the necessary attributes. There is no difference in the way items are treated in these two types of organizations except that there cannot be any financial transactions in the downstream applications for items that are assigned to an item organization.

Item Master Organization: Explained

An item master organization lists and describes items that are shared across several inventory organizations or item organization.

The following example shows the choice between inventory organizations that track inventory transactions, stored in two warehouses, and item organizations that just track items, listed in two sales catalogs.
For the most efficient processing, you should:

- Have a single item master
- Include an item and its definition of form, fit, and function only once in the item master
- Separate the item master organization from organizations that store and transact items

Note
Oracle Fusion allows multiple item masters, however, use this capability cautiously. If you acquire a company, there may be value in allowing the old item master to exist for a transition period. If you manage your subsidiaries as separate businesses, there may be reduced value in a single item master.

Define Social Networking

How can I enable social networking features?

Set the Social Networking Features Enabled profile option to “Y” on the Manage Social Networking Profile Option Values page. If you enable this profile option, users can access the following features:

- Kudos
- Message board
- Activity stream
- Connections
- Personal status

Note
If you disable social networking features, users can continue to access the bookmarks they created from the Bookmarks region in the Activities and Interests card on the My Portrait tab. Also, the spaces that users are a member of continue to appear in the Spaces region.

Define Workforce Profiles

Profile Management: Explained

Profile management provides a framework for developing and managing talent profiles that meet your industry or organizational requirements. Profiles summarize the qualifications and skills of a person or a workforce structure.
such as a job or position. Profiles are valuable for tracking workers' skills, competencies, and accomplishments, and for various talent management activities, such as career planning, identifying training needs, performance management, and in the recruitment process for identifying job requirements and suitable applicants.

This topic discusses:

- Profile search
- Profile comparison
- Best-fit analysis

**Profile Search**

You can search profiles for workers, jobs, and positions with certain criteria. For example, an HR (Human Resources) specialist in London who is looking to fill an applications developer position from within the company can search for profiles of all workers who are based in London and have experience with Java and PL/SQL.

**Profile Comparison**

Using the comparison feature, you can compare profiles to determine next career moves or training needs for workers, and identify suitable candidates for jobs. For example, if John is looking for his next career move, he can compare his profile to that of a job to determine whether his competency ratings match the targeted competency ratings in a job profile. For example, if his Teamwork rating is 3 and the Product Strategy Teamwork requirement is 4, he has a deficiency of -1. John and his manager can use this gap to drive development plans and for other talent management-related functions.

**Best-Fit Analysis**

Use the best-fit analysis to determine the person profile that most closely matches a job profile, or the job profile that is the best match for a person profile. For example, if you are trying to fill a Developer vacancy, and the job profile requires a B.S. degree in Computer Science, level 4 expertise coding Java, and a Teamwork rating of at least 3, you can review an automatically-generated list of workers who most closely match this set of requirements. You can also use the best-fit analysis to find workers who are similar to a selected worker, or jobs that are similar to a selected job.

**Oracle Fusion Profile Management Components: How They Work Together**

You can configure Oracle Fusion Profile Management to meet your business requirements using these components: the content library, profiles and profile types, content subscribers, educational establishments, instance qualifier sets, and rating models.

This figure illustrates how the components of Profile Management fit together.
Content Library

The content library provides the foundation for profiles as it stores both content types and content items.

Profile Types

Profile types are templates that you use to create profiles. Profile types determine whether the profile is for a person or for a workforce structure such as a job or a position, and the content of the profile. You select content types from the content library to create content sections for the profile type.

Profiles

You create person profiles for individual workers and model profiles for workforce structures, such as a jobs or positions. The information that you complete for the profile is determined by how the profile type has been set up. For example, a person profile might contain information about a person’s education, language skills, competencies, and activities and interests. A job profile might contain information about the requirements for the job, such as competencies, language skills, degrees, or certifications.
Content Subscribers
Content subscribers are applications external to Oracle Fusion Profile Management that use content types.

Educational Establishments
You can define educational establishments for workers to use when they add education information, such as degrees, to their profile.

Instance Qualifier Sets
You assign instance qualifiers to content types. Instance qualifier sets uniquely identify multiple instances of a content item on a profile. For example, if multiple people update a performance rating for a competency on a worker’s profile, instance qualifiers provide a unique identifier to each instance of the competency so that you can determine who provided each rating.

Rating Models
When you create content types in the content library, you can attach rating models to determine the scale for measuring performance and proficiency. You can also use rating models to measure the risk and impact of loss for workers, and to measure their potential.

Oracle Fusion Profile Management, Performance Management, Goal Management, and Talent Review: How They Work Together
Oracle Fusion Profile Management supports talent management business processes in these products:
- Oracle Fusion Performance Management
- Oracle Fusion Goal Management
- Oracle Fusion Talent Review

Oracle Fusion Performance Management
Oracle Fusion Performance Management uses the rating models that you define in Profile Management to rate workers on their performance. When you define a performance document template, you can specify whether the ratings and comments from managers and workers are uploaded automatically to workers’ profiles when the performance document is finalized. Instance qualifier sets distinguish the manager ratings from the workers’ self ratings. Performance Management also uses competencies from the content library in performance documents.

Oracle Fusion Goal Management
You can set up a content type relationship between the Goals content type and other content types, such as the Competencies content type and the Memberships content type. Using these relationships, you can then set up target outcomes for goals. Target outcomes are the content items within the content type that is related to the Goals content type. For example, if you set up a relationship between the Goals content type and the Competencies content type, workers can add a target outcome of a specific competency to their goals. In this case, the specific competency is the content item within the Competencies content type. When workers complete the goal, their profiles are updated to include the competency.

Oracle Fusion Talent Review
Oracle Fusion Talent Review uses information from the Performance and Potential and Risk of Loss sections within a worker’s profile to build the analytics that are part of the talent review process. These sections are defined as
content types within the content library and included in the person profile type. When a talent review is complete, workers’ profiles are updated automatically with the performance rating given during calibration discussions. Instance qualifier sets enable you to distinguish the talent review rating from ratings given by the worker’s manager, a peer, or perhaps the worker’s self-evaluation.

Define Talent Profile Settings

Profile Management Lookups: Explained

This topic identifies common lookups that are profile management-related and have user or extensible customization levels. Review these lookups, and update them as appropriate to suit enterprise requirements.

Profile Management Lookups

Profile management lookups are described in the following table.

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Description</th>
<th>Customization Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRT_CONTENT_SUPP_CODE</td>
<td>Codes to identify where the content type originated, such as from the customer or from a third party. HR specialists assign supplier codes when setting up new content types.</td>
<td>Extensible</td>
</tr>
<tr>
<td>HRT_COMPETENCY_EVAL_TYPE</td>
<td>Types of competency ratings, such as self, supervisor, and peer. Evaluation types are used in instance qualifier sets to identify the role of the person who provided a competency rating for a worker.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_DEGREE_LEVEL</td>
<td>Required levels of education for a job or position profile, such as a Bachelor’s Degree or Masters degree.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_DURATION</td>
<td>Time periods for when a worker would like to make a career move. Workers select a duration for their career move in the career preferences information on the career planning card.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_EDUCATION_LEVEL</td>
<td>Types of education that a degree provides, such as a Bachelor’s degree or one year college. HR specialists use education levels when creating content items in the Degrees content type.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_IMPORTANCE</td>
<td>Values for the importance level of each content type. Importance values are used in the best-fit analysis calculation.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_PAY_FREQUENCY</td>
<td>Frequencies at which a worker is paid. Workers specify a preferred pay frequency when they edit their work requirements on the experience and qualifications card.</td>
<td>User</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>HRT_READINESS</td>
<td>Time periods in which a worker will be ready for advancement to a specified job or position profile. Managers and human resource (HR) specialists use readiness levels when editing the talent ratings information on workers’ career planning cards. User</td>
<td></td>
</tr>
<tr>
<td>HRT_RISK_REASON</td>
<td>Reasons for why a worker is at risk for leaving the company. Managers and HR specialists use risk reasons when editing the talent ratings information on workers’ career planning cards. User</td>
<td></td>
</tr>
<tr>
<td>HRT_SATISFACTION_METHOD</td>
<td>Methods by which an item such as a competency is verified, such as assessment or demonstration. Managers and HR specialists use these values to identify the method that was used to evaluate the proficiency of a worker in a particular skill or quality. User</td>
<td></td>
</tr>
<tr>
<td>HRT_TRAVEL_FREQUENCY</td>
<td>Percentages of time a worker is expected to travel for a job or position, for example, 25 percent or 50 percent. Managers and HR specialists specify the required travel frequency on job and position profiles. Workers also specify a travel frequency when they edit their work requirements on the experience and qualifications card. User</td>
<td></td>
</tr>
<tr>
<td>HRT_VALIDITY_PERIOD</td>
<td>Periods of time for which content items such as licenses and certifications are valid. Managers and human resource (HR) specialists select validity periods when setting up content items for the Licenses and Certifications content type. User</td>
<td></td>
</tr>
<tr>
<td>HRT_WORK_DAYS</td>
<td>Work days for a job or position, such as Monday through Friday. Managers and HR specialists specify the expected work days on job and position profiles. User</td>
<td></td>
</tr>
<tr>
<td>HRT_WORK_HOURS</td>
<td>Work hours for a job or position, such as 9:00 a.m. to 5:00 p.m. Managers and HR specialists specify the expected work hours on job and position profiles. User</td>
<td></td>
</tr>
</tbody>
</table>

**Value Sets: Explained**

A value set is a set of valid values that you assign to a flexfield segment.
An end user enters a value into a flexfield segment while using the application. The flexfield validates the segment against the set of valid values that you configured as a value set and assigned to the segment. For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

**Caution**

Be sure changes to a shared value set are compatible with all flexfields segments using the value set.

Defining value sets involves making decisions about the following.

- Validation
- Security
- Precision and scale
- Usage and deployment

**Validation**

The following types of validation are available for value sets.

- Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set does not present a list of valid values to users.

You can build a tree structure from the values in an independent value set whose data type is character.

**Note**

Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

For more information, see the Oracle Fusion Applications Extensibility Guide.

**Security**

Value set security only works in conjunction with usage within flexfield segments. If a value set is used standalone, meaning outside a flexfield, value set security is not applied, but Oracle Fusion data security is enforced.

You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

Value set security applies at the value set level. If a value set is secured, every usage of it in any flexfield is secured. It is not possible to disable security for individual usages of the same value set.

Value set security applies to independent, dependent or table-validated value sets.
Value set security applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security does not determine which descriptive flexfield data is shown upon querying.

Security conditions defined on value sets will always use table aliases. When filters are used, table aliases are always used by default. When predicates are defined for data security conditions, make sure that the predicates will also use table aliases.

For key flexfields, the attributes in the view object that correspond to the code combination ID (CCID), structure instance number (SIN) and data set number (DSN) cannot be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the CCID segment is the common attribute.

**Precision and Scale**

For a value set with the data type Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

**Usage and Deployment**

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.

The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.
For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

### Defining Value Sets: Critical Choices

Validation and usage of value sets determine where and how end users access valid values for attributes represented by flexfield segments.

**Tip**

As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

### Value Sets for Context Segments

When assigning a value set to a context segment, you can only use table-validated or independent value sets. The data type must be character and the maximum length of the values being stored must not be larger than column length of the context.

### Format Only Validation

The format only validation type enables end users to enter any value, as long as it meets your specified formatting rules. That is, the value must not exceed the maximum length you define for your value set, and it must meet any format requirements for that value set.

For example, if the value set allows only numeric characters, your user could enter the value 456 (for a value set with maximum length of three or more), but could not enter the value ABC. A format only value set does not otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how may digits should follow the radix separator.

### Interdependent Value Sets

You cannot specify a dependent value set for a given segment without having first defined an independent value set that you apply to another segment in the same flexfield. You use a dependent value set to limit the list of values for a given segment based on the value that the end user has chosen for a related independent segment. The available values in a dependent list and the meaning of a given value depend on which value was selected for the independently validated segment.

For example, you could define an independent value set of U.S. states with values such as CA, NY, and so on. Then you define a dependent value set of U.S. cities, with values such as San Francisco and Los Angeles that are valid for the independent value CA, and New York City and Albany that are valid for the independent value NY. In the UI, only the valid cities can be selected for a given state.
Because you define a subset value set from an existing independent value set, you must define the independent value set first. End users do not need to choose a value for another segment first to have access to the subset value set.

Table Validation

Typically, you use a table-validated set when the values you want to use are already maintained in an application table (for example, a table of vendor names). Table validation allows you to enable a segment to depend upon multiple prior segments in the same context or structure.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it does not have bind variables. If it has bind variables, the assumption is that the values are unique in the value set.

Range

In the case of format, independent, or dependent value sets, you can specify a range to further limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of segments where one segment represents the low end of the range and another segment represents the high end of the range.

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100. If you use a table value set, you cannot reference flexfield segments in the WHERE clause of the value set. For example, the WHERE clause cannot reference a segment or a value set.

Security

In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

When you enable security on a table-validated value sets, the security rule that is defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set should not assume anything about the bind variables; it must assume the whole list of values is available and write the rule, for example, to allow x, or to allow y and z. By default in data security all values are denied, and show only rows to which access has been provided.

Maintenance

There is no need to define or maintain values for a table-validated or subset value set, as the values are managed as part of the referenced table or independent value set, respectively.

If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select **Translatable**. This does not require you to provide translated values now, but you cannot change this option if you decide to provide them later.
For more information about defining value sets, see the Oracle Fusion Applications Extensibility Guide.

**Define Talent Profile Settings**

**Profile Management Lookups: Explained**

This topic identifies common lookups that are profile management-related and have user or extensible customization levels. Review these lookups, and update them as appropriate to suit enterprise requirements.

**Profile Management Lookups**

Profile management lookups are described in the following table.

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Description</th>
<th>Customization Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRT_CONTENT_SUPP_CODE</td>
<td>Codes to identify where the content type originated, such as from the customer or from a third party. HR specialists assign supplier codes when setting up new content types.</td>
<td>Extensible</td>
</tr>
<tr>
<td>HRT_COMPETENCY_EVAL_TYPE</td>
<td>Types of competency ratings, such as self, supervisor, and peer. Evaluation types are used in instance qualifier sets to identify the role of the person who provided a competency rating for a worker.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_DEGREE_LEVEL</td>
<td>Required levels of education for a job or position profile, such as a Bachelor’s Degree or Masters degree.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_DURATION</td>
<td>Time periods for when a worker would like to make a career move. Workers select a duration for their career move in the career preferences information on the career planning card.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_EDUCATION_LEVEL</td>
<td>Types of education that a degree provides, such as a Bachelor’s degree or one year college. HR specialists use education levels when creating content items in the Degrees content type.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_IMPORTANCE</td>
<td>Values for the importance level of each content type. Importance values are used in the best-fit analysis calculation.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_PAY FREQUENCY</td>
<td>Frequencies at which a worker is paid. Workers specify a preferred pay frequency when they edit their work requirements on the experience and qualifications card.</td>
<td>User</td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
<td>User</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>HRT_READINESS</td>
<td>Time periods in which a worker will be ready for advancement to a specified job or position profile. Managers and human resource (HR) specialists use readiness levels when editing the talent ratings information on workers’ career planning cards.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_RISK_REASON</td>
<td>Reasons for why a worker is at risk for leaving the company. Managers and HR specialists use risk reasons when editing the talent ratings information on workers’ career planning cards.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_SATISFACTION_METHOD</td>
<td>Methods by which an item such as a competency is verified, such as assessment or demonstration. Managers and HR specialists use these values to identify the method that was used to evaluate the proficiency of a worker in a particular skill or quality.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_TRAVEL FREQUENCY</td>
<td>Percentages of time a worker is expected to travel for a job or position, for example, 25 percent or 50 percent. Managers and HR specialists specify the required travel frequency on job and position profiles. Workers also specify a travel frequency when they edit their work requirements on the experience and qualifications card.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_VALIDITY_PERIOD</td>
<td>Periods of time for which content items such as licenses and certifications are valid. Managers and human resource (HR) specialists select validity periods when setting up content items for the Licenses and Certifications content type.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_WORK_DAYS</td>
<td>Work days for a job or position, such as Monday through Friday. Managers and HR specialists specify the expected work days on job and position profiles.</td>
<td>User</td>
</tr>
<tr>
<td>HRT_WORK_HOURS</td>
<td>Work hours for a job or position, such as 9:00 a.m. to 5:00 p.m. Managers and HR specialists specify the expected work hours on job and position profiles.</td>
<td>User</td>
</tr>
</tbody>
</table>

**Value Sets: Explained**

A value set is a set of valid values that you assign to a flexfield segment.
An end user enters a value into a flexfield segment while using the application. The flexfield validates the segment against the set of valid values that you configured as a value set and assigned to the segment. For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue. Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

**Caution**

Be sure changes to a shared value set are compatible with all flexfields segments using the value set.

Defining value sets involves making decisions about the following.

- Validation
- Security
- Precision and scale
- Usage and deployment

**Validation**

The following types of validation are available for value sets.

- Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set does not present a list of valid values to users.

You can build a tree structure from the values in an independent value set whose data type is character.

**Note**

Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

For more information, see the Oracle Fusion Applications Extensibility Guide.

**Security**

Value set security only works in conjunction with usage within flexfield segments. If a value set is used standalone, meaning outside a flexfield, value set security is not applied, but Oracle Fusion data security is enforced. You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

Value set security applies at the value set level. If a value set is secured, every usage of it in any flexfield is secured. It is not possible to disable security for individual usages of the same value set.

Value set security applies to independent, dependent or table-validated value sets.
Value set security applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security does not determine which descriptive flexfield data is shown upon querying.

Security conditions defined on value sets will always use table aliases. When filters are used, table aliases are always used by default. When predicates are defined for data security conditions, make sure that the predicates will also use table aliases.

For key flexfields, the attributes in the view object that correspond to the code combination ID (CCID), structure instance number (SIN) and data set number (DSN) cannot be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the CCID segment is the common attribute.

**Precision and Scale**

For a value set with the data type Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

**Usage and Deployment**

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.

The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.
For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

Defining Value Sets: Critical Choices

Validation and usage of value sets determine where and how end users access valid values for attributes represented by flexfield segments.

Tip
As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

Value Sets for Context Segments

When assigning a value set to a context segment, you can only use table-validated or independent value sets. The data type must be character and the maximum length of the values being stored must not be larger than column length of the context.

Format Only Validation

The format only validation type enables end users to enter any value, as long as it meets your specified formatting rules. That is, the value must not exceed the maximum length you define for your value set, and it must meet any format requirements for that value set.

For example, if the value set allows only numeric characters, your user could enter the value 456 (for a value set with maximum length of three or more), but could not enter the value ABC. A format only value set does not otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how many digits should follow the radix separator.

Interdependent Value Sets

You cannot specify a dependent value set for a given segment without having first defined an independent value set that you apply to another segment in the same flexfield. You use a dependent value set to limit the list of values for a given segment based on the value that the end user has chosen for a related independent segment. The available values in a dependent list and the meaning of a given value depend on which value was selected for the independently validated segment.

For example, you could define an independent value set of U.S. states with values such as CA, NY, and so on. Then you define a dependent value set of U.S. cities, with values such as San Francisco and Los Angeles that are valid for the independent value CA, and New York City and Albany that are valid for the
independent value NY. In the UI, only the valid cities can be selected for a given state.

Because you define a subset value set from an existing independent value set, you must define the independent value set first. End users do not need to choose a value for another segment first to have access to the subset value set.

**Table Validation**

Typically, you use a table-validated set when the values you want to use are already maintained in an application table (for example, a table of vendor names). Table validation allows you to enable a segment to depend upon multiple prior segments in the same context or structure.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it does not have bind variables. If it has bind variables, the assumption is that the values are unique in the value set.

**Range**

In the case of format, independent, or dependent value sets, you can specify a range to further limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of segments where one segment represents the low end of the range and another segment represents the high end of the range.

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100. If you use a table value set, you cannot reference flexfield segments in the WHERE clause of the value set. For example, the WHERE clause cannot reference a segment or a value set.

**Security**

In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

When you enable security on a table-validated value sets, the security rule that is defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set should not assume anything about the bind variables; it must assume the whole list of values is available and write the rule, for example, to allow x, or to allow y and z. By default in data security all values are denied, and show only rows to which access has been provided.

**Maintenance**

There is no need to define or maintain values for a table-validated or subset value set, as the values are managed as part of the referenced table or independent value set, respectively.
If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select **Translatable**. This does not require you to provide translated values now, but you cannot change this option if you decide to provide them later.

For more information about defining value sets, see the Oracle Fusion Applications Extensibility Guide.

**Define Talent Profile Content**

**Rating Models: Explained**

Use rating models to rate workers on their performance and level of proficiency in the skills and qualities that are set up on the person profile. You can also use rating models to specify target proficiency levels for items on a model profile, so that the model profile can be compared to workers' profiles.

To rate workers on their performance and proficiency, you attach rating models to the content types that are included in the person profile, and then workers can be rated on the items within the type. For example, you can rate workers on the Communication content item within the Competencies content type.

For model profiles, you can specify target proficiency levels for items on the profile, so that the model profile can be compared to workers' profiles. Using the ratings, managers can compare a model profile to workers' profiles to determine the best person suited to fill a position. Workers can compare their profile to model profiles to identify other positions within the organization that they are suited for, or to identify gaps in skills that they need to fill before applying for other positions.

Rating models that measure workers' potential and the impact and risk of loss are also available.

Rating models can include some or all of the following components, depending on the use for the model:

- Rating levels
- Review points
- Rating categories
- Distributions

**Rating Levels**

Rating levels identify the qualitative values, such as 1, 2, 3, or 4, that you use to rate or score a worker's performance. Define numeric ratings for rating models that you use with performance documents that use the average calculation method.

**Review Points**

Define review points for rating models that you use with performance documents that use the sum or band calculation method. The review points and point ranges that you define for the rating model are used to calculate ratings.
Rating Categories

Rating categories enable you to group rating levels together for analysis tools used in the talent review process, such as the box chart that is used in the talent review process. You can group rating levels into categories such as low, medium, and high, and those categories then become the labels for the analytic. You should not change rating categories after setting them up, as the changes could affect the analytic.

Distributions

Oracle Fusion Compensation Management and Oracle Fusion Performance Management both use rating model distributions to determine the targeted minimum and maximum percentage of workers that should be given each rating level. Compensation Management uses the distribution values that you set up directly on rating models. However, you can set up distributions at the performance template level for rating models that are used in Performance Management.

Content Types: Explained

Content types are the skills, qualities, and qualifications that you want to track in talent profiles. The content library contains predefined content types such as competencies, languages, and degrees, but you can create new content types as needed. You can also create free-form content types.

Content types contain:

- Properties
- Relationships
- Subscribers

Note

Free-form content types do not contain relationships and do not contain properties until you add them to a profile type.

Properties

For each content type, you define the properties that all content items of the content type can or must have. To define properties of the content type, you select fields to be displayed when setting up the content items and the attributes of those fields. The attributes that you specify for each field are: field label, default value, whether the field is required, and whether the field is hidden, display-only, or editable. If the field is attached to a predefined list of values, you also specify the source of the list.

Relationships

Specify where one content type is a parent of another, or where one content type supports another. Content items of content types with relationships inherit the relationship. You cannot create two kinds of relationships between two types or create a relationship between a type and itself. For example, content type A
cannot be both the parent and child of content type B. A content type cannot be related to itself.

**Subscribers**

Specify the subscriber codes of the applications or other Oracle Fusion products that use each content type. If you do not specify a subscriber code for the content type, you cannot view the content type in other applications. For example, if you add a new content type called Corporate Citizenship to the person profile type, you cannot view the content section for Corporate Citizenship in person profiles until you add the new content type to the HRMS content subscriber code.

**Content Type Relationships: Examples**

Content relationships enable you to associate content items of related content types with each other. The following scenarios illustrate the use of content type relationships.

**Tracking Product Expertise**

The Resource Manager component of Oracle Fusion Trading Community Model uses content type relationships to track the areas of expertise of workers. Using the predefined content type relationship where the Categories content type is a parent of Products, and Products is a parent of Components, resource managers can keep track of the categories, products, and components that are considered to be their areas of expertise for their resources.

---

**Note**

Because these content types and relationships are applicable only to the Resource Manager component of Oracle Fusion Trading Community Model, this product is the only predefined content subscriber to these content types.

---

**Specifying Target Outcomes for Goals**

To help your workers manage their goals, you want them to associate their goals with target outcomes, which are content types such as Competencies and Memberships. To accomplish this, you can set up a relationship on the Competencies content type where Competencies is supported by Goals. Workers can then set up goals that have a specific competency as a target outcome.

**Content Items: Explained**

Content items are the individual skills, qualities, and qualifications within the content types in the content library. For example, within the Competencies content type, communication is a content item. You can create content items to meet your business needs.

This topic discusses:

- Item properties
- Related content items
- Proficiency descriptions
Item Properties

Content items inherit the fields and field properties that you define for the content type to which the item belongs. For example, one of the fields defined for the Memberships content type is ITEM_DESCRIPTION field. The attributes of this field are set up so that the label is Description, the field is editable, and the field does not require an entry. When you set up a content item for the Memberships content type, you will see a field labeled Description, in which you can enter text to describe the agency, but the field will not be required.

Related Content Items

If the content type for which you are creating an item has related content types, then you can enter the related content items for the item. For example, if you have a content type relationship where the Competencies content type is supported by the Goals content type, then on the content items for competencies, you can enter the related goals.

Proficiency Descriptions

If the content item belongs to a content type that has a rating model defined for it, then you can either use the existing descriptions for the ratings within the model, or define descriptions for the ratings that are specific to the content item. When ratings are given for the content item, the descriptions defined for the item are used instead of those on the rating model.

Creating Content Types and Content Items: Worked Example

This example demonstrates how to set up a new content type and content items to track the corporate citizenship activities of your workers so that you can rate them on their involvement in the organization. This example also demonstrates how to set up a rating model to be used with the content type and add the new content type to the person profile.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can a predefined rating model be used to rate corporate citizenship?</td>
<td>No. The predefined rating models do not have relevant rating descriptions.</td>
</tr>
<tr>
<td>Should the content type be a free-form content type?</td>
<td>No. Content items are needed, and the content should be stored in the content library.</td>
</tr>
<tr>
<td>What field and properties should the content type contain?</td>
<td>Add two fields to the content type:</td>
</tr>
<tr>
<td></td>
<td>• ITEM_TEXT1</td>
</tr>
<tr>
<td></td>
<td>• RATING_MODEL_ID</td>
</tr>
</tbody>
</table>

The ITEM_TEXT1 field will have a label of Comments, and will be used to enter comments about the workers’ corporate involvement. The RATING_MODEL_ID field will have a label of Company Contribution and will be used to attach the rating model for corporate citizenship to the content type.

Both fields should require entry and should be editable.
Does the content type need any content subscribers? | Yes. In order to be visible on the person profile, the new content type must be added to the HRMS content subscriber code.
---|---
What content items are needed to track the required information? | • Corporate social responsibility
• Corporate environmental responsibility
• Corporate industrial citizenship
• Corporate state citizenship
• Corporate borough, council, or municipal citizenship
---|---
When the content type is added to the person profile as a content section, what properties should the fields contain? | Both the Comments field and the Company Contribution fields should display in the detail view of the content section, they should be required, and they should be included in search results.

To track corporate citizenship for your workers, complete the following tasks:

• Create a rating model.
• Create a content type.
• Create content items.
• Add the content type to the person profile type.

Creating a Rating Model

1. In the Setup and Maintenance work area, search for the Manage Profile Rating Models task and click Go to Task.
2. On the Manage Rating Models page, click Create.
3. On the Create Rating Model page, complete the following fields, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Citizenship</td>
</tr>
<tr>
<td>Rating Name</td>
<td>Corporate Citizenship</td>
</tr>
<tr>
<td>Description</td>
<td>Rating model for corporate citizenship</td>
</tr>
</tbody>
</table>
4. On the Rating Levels tab, complete the following fields, as shown in this table.

<table>
<thead>
<tr>
<th>Rating Level</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Demonstrates limited or unused influence.</td>
</tr>
<tr>
<td>2</td>
<td>Demonstrates clear evidence of influence.</td>
</tr>
<tr>
<td>3</td>
<td>Provides a successful image of the company as socially responsible in limited environments.</td>
</tr>
</tbody>
</table>
4. Actively called upon to use influence as a corporate representative in selected environments.

5. Demonstrates high level of influence and is able to operate effectively in all environments.

5. Click **Save and Close**.

**Creating a Content Type**

1. In the Setup and Maintenance work area, search for the **Manage Profile Content Types** task and click **Go to Task**.

2. On the Manage Content Types page, click **Create**.

3. On the Create Content Type page, add a content type by completing the following fields, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Citizenship</td>
</tr>
<tr>
<td>Name</td>
<td>Corporate Citizenship</td>
</tr>
<tr>
<td>Description</td>
<td>Ratings for corporate citizenship behaviors for workers.</td>
</tr>
</tbody>
</table>

4. Set up the following field properties, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Label</th>
<th>Required</th>
<th>Display Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_TEXT_1</td>
<td>Comments</td>
<td>Selected</td>
<td>Editable</td>
</tr>
<tr>
<td>RATING_MODEL_</td>
<td>Company Contribution</td>
<td>Selected</td>
<td>Editable</td>
</tr>
</tbody>
</table>

5. Click **Save and Close**.

6. On the Manage Content Types page, select the **Corporate Citizenship** content type and click **Edit**.

7. On the Edit Content Type page, select the **Subscribers** tab.

8. On the Subscribers tab, click **Add**.

9. In the Subscriber Code field, select **HRMS**.

10. Click **Save and Close**.

**Creating Content Items**

1. In the Setup and Maintenance work area, search for the **Manage Profile Content Items** task and click **Go to Task**.

2. On the Manage Content Items page, click **Create**.

3. In the Create Content Item dialog box, complete the following fields, as shown in this table.
4. On the Create Content Item: Corporate Social Responsibility page, select the Corporate Citizenship rating model in the Rating field.

5. Click Save and Close.

6. Repeat steps 2 through 5 to add content items for Corporate Environmental Responsibility, Corporate Industrial Citizenship, Corporate State Citizenship, and Corporate Borough, Council, or Municipal Citizenship.

### Adding the Corporate Citizenship Content Type to the Person Profile Type

1. In the Setup and Maintenance work area, search for the Manage Profile Types task and click Go to Task.

2. On the Manage Profile Types page, locate the Person profile type and click Edit.

3. On the Edit Profile Type: Person page, select the Content Sections tab.

4. In the Content Sections region, click Add Content Section.

5. In the Content Types dialog box, select Citizenship.

6. In the Content Sections region, click Citizenship.

7. On the Content Section page, set up the following field properties, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Display Flag</th>
<th>Required</th>
<th>Searchable</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_DESCRIPTION</td>
<td>Detail</td>
<td>Selected</td>
<td>Selected</td>
</tr>
<tr>
<td>RATING_LEVEL_ID1</td>
<td>Detail</td>
<td>Selected</td>
<td>Selected</td>
</tr>
</tbody>
</table>

8. Click OK.

9. On the Edit Profile Type: Person page, click Save and Close.

### Free-Form Content Types: Worked Example

This example demonstrates how to set up a free-form content type, add it to the HRMS content subscriber code, and then add the content type to the person profile type.

Your company wants to track the previous employment information for workers, including employer name, dates of employment, and job description. However, you do not want to set up and maintain content items for each employer, and this information applies only to person profiles. You decide to use a free-form content type for this information. You can set up the free-form content type with minimal information, and then when you add it to the person profile as a content section, you can define properties for employer name, dates of employment,
and job description. Workers can complete their employment information on their profile based on how you set up the content section. The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Should the content type be a free-form content type?</td>
<td>Yes. Content items are not needed, and the content should not be stored in the content library.</td>
</tr>
<tr>
<td>Does the content type need any content subscribers?</td>
<td>Yes. In order to be visible on the person profile, the new content type must be added to the HRMS content subscriber code.</td>
</tr>
<tr>
<td>When the content type is added to the person profile as a content section, what fields are needed?</td>
<td>To capture the previous employer, a text field is needed. To capture employment dates, two date fields are needed. To capture job description, another text field is needed. Therefore, the following fields must be added:</td>
</tr>
<tr>
<td></td>
<td>• ITEM_TEXT30_1</td>
</tr>
<tr>
<td></td>
<td>• ITEM_DATE_1</td>
</tr>
<tr>
<td></td>
<td>• ITEM_DATE_2</td>
</tr>
<tr>
<td></td>
<td>• ITEM_TEXT240_1</td>
</tr>
<tr>
<td>What properties are needed for the ITEM_TEXT30_1 field?</td>
<td>The field should not be required and the information should not be available as search criteria, so you need only set up these field properties as follows:</td>
</tr>
<tr>
<td></td>
<td>• Label: Previous Employer</td>
</tr>
<tr>
<td></td>
<td>• Display: Detail (section should appear in detail view of profile)</td>
</tr>
<tr>
<td>What properties are needed for the ITEM_DATE_1 field?</td>
<td>The field should not be required and the information should not be available as search criteria, so you need only set up these field properties as follows:</td>
</tr>
<tr>
<td></td>
<td>• Label: From Date</td>
</tr>
<tr>
<td></td>
<td>• Display: Detail</td>
</tr>
<tr>
<td>What properties are needed for the ITEM_DATE_2 field?</td>
<td>The field should not be required and the information should not be available as search criteria, so you need only set up these field properties as follows:</td>
</tr>
<tr>
<td></td>
<td>• Label: To Date</td>
</tr>
<tr>
<td></td>
<td>• Display: Detail</td>
</tr>
<tr>
<td>What properties are needed for the ITEM_TEXT240_1 field?</td>
<td>The field should not be required and the information should not be available as search criteria, so you need only set up these field properties as follows:</td>
</tr>
<tr>
<td></td>
<td>• Label: Job Description</td>
</tr>
<tr>
<td></td>
<td>• Display: Detail</td>
</tr>
<tr>
<td>What role access should be granted for the content section?</td>
<td>Employees, managers, and HR specialists should all have access to update the section.</td>
</tr>
</tbody>
</table>

To set up a free-form content type to track previous employment information for workers, you must:
• Set up a free-form content type
• Add the free-form content type to the person profile type

Setting Up a Free-Form Content Type

1. In the Setup and Maintenance work area, search for the Manage Profile Content Types task and click Go to Task.
2. On the Manage Content Types page, click Create.
3. On the Create Content Type page, complete the following fields, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>PREVEMP</td>
</tr>
<tr>
<td>Name</td>
<td>Previous Employment</td>
</tr>
<tr>
<td>Description</td>
<td>Track previous employment information for workers.</td>
</tr>
</tbody>
</table>

4. Select the Free-Form Type check box.
5. Click Save and Close.
6. On the Manage Content Types page, select the Previous Employment content type and click Edit.
7. On the Edit Content Type page, select the Subscribers tab.
9. Click Save and Close.

Adding the Free-Form Content Type to the Person Profile Type

1. In the Setup and Maintenance work area, search for the Manage Profile Types task and click Go to Task.
2. On the Manage Profile Types page, select the Person profile type, and click Edit.
3. On the Edit Profile Type: Person page, select the Content Sections tab and click Add Content Section.
4. In the Content Types dialog box, select the Previous Employment content type.
5. In the Content Sections region, click the Previous Employment content type and enter the following properties on the Content Section page, as shown in this table.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Label</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_TEXT30_1</td>
<td>Previous Employer</td>
<td>Detail</td>
</tr>
<tr>
<td>ITEM_DATE_1</td>
<td>From Date</td>
<td>Detail</td>
</tr>
<tr>
<td>ITEM_DATE_2</td>
<td>To Date</td>
<td>Detail</td>
</tr>
<tr>
<td>ITEM_TEXT240_1</td>
<td>Job Description</td>
<td>Detail</td>
</tr>
</tbody>
</table>
6. In the Content Access Section region, click Add.
7. In the Role field, select Employee.
8. Select the Update check box.
9. Click Add.
10. In the Role field, select Manager.
11. Click OK.
12. Click Add.
13. In the Role field, select HR Specialist.
14. Click OK.
15. On the Edit Profile Type: Person page, click Save and Close.

FAQs for Define Talent Profile Content

**What's a rating category?**

A label for a grouping of rating levels. Rating categories are used in talent management processes such as performance management and talent reviews to group ratings for analysis tools such as the performance and potential box chart.

**Why are some content type relationships not editable?**

You can edit any content type relationships that you define. However, the relationships that are predefined cannot be changed.

**How can I define a relationship between the Goals content type and other content types?**

Set up the relationship on the content type that you want to relate to goals using the relationship type: Is supported by. For example, if you want to define a relationship between the Goals content type and the Competencies content type, set up the relationship on the Competencies content type, instead of the Goals content type.

**What's a free-form content type?**

Free-form content types enable you to capture information in a profile that you do not need to store in the content library. For example, you can set up a free-form content type to store information about the previous employment information for your workers.

A free-form content type contains only a code, name, and a description, and does not have any properties defined for it until you add it to a profile type. Free-form content types do not include any content items.

**Why can't I change the relationship type of a content item?**

A content item's relationship type is derived from its content type, and you cannot change it. You can only change relationships at the content type level. You cannot change predefined relationships.
Define Talent Profiles

Profile Types: Explained

Profile types include person profile types and model profile types. The person profile type is the template that you use to create profiles of your workers. The person profile contains the skills, qualities, and qualifications that you want to track for your workers. The person profile type is predefined, and you can have only one. Model profile types are templates for workforce structures such as jobs and positions. Model profiles identify the targeted and required skills and qualifications for a job or position, and also identify work requirements, such as work schedule and travel frequency. You can set up multiple model profile types.

To define profile types, you first specify whether the profile type is a person or model profile. For model profiles, you also specify the workforce structures for which the model profile can be used. For example, if you specify that the model profile can be used for jobs and positions, then you can use the profile type to create both job and position profiles. To define the structure of the profile type, you add one or more content sections using content types from the content library and free-form content types. Define the following for each content section:

- Instance qualifier sets
- Section properties
- Role access

Instance Qualifier Sets

If you have defined instance qualifier sets for the content type, you select the instance qualifier set to use for the sections.

Section Properties

The properties determine the fields and how they are displayed when you create profiles based on the type. For example, properties determine the label for the field, whether the field is required, and whether the field should be included in profile searches. For sections with content types from the content library, you can use the field properties as they have been defined in the content library, or add, remove, or change the properties to suit the content section. You define all of the properties for free-form content types.

Role Access

You can specify the user roles, such as Employee or Manager, that can view the content section, and which user roles have access to update the section.

Instance Qualifier Sets: Explained

An instance qualifier set is a group of codes that you use to uniquely identify different occurrences of the same profile item within the Competency content type. Instance qualifiers typically identify the role of the person who edited a competency. For example, if a worker, the worker's peer, and the worker's manager all enter a rating for a competency on the worker's profile, instance qualifier sets uniquely identify each instance, or, the rating given by each different role. Uniquely identifying different instances of competencies enables you to specify which instance is used when you view or compare profiles.
Each instance qualifier contains a code and a description, which indicate the role or the application that updated the competency. For example, P is the code that is used when an employee’s peer rates the employee and T is used for the rating that results from the talent review meeting. You can use the predefined codes and descriptions, or you can create your own.

In addition to the code and description, each instance qualifier has the following properties:

- Priority
- Employer and manager views
- Search ability
- Default instance qualifier for employer and manager

**Priority**

Priority determines the order in which different instances of a competency are displayed, and also determines which instance to use when searching and comparing profiles. The lowest number indicates the highest priority.

**Employer and Manager Views**

Employer and manager views determine which instances are visible to employees and to managers.

**Search Ability**

You can specify whether items that have been assigned the instance qualifier code should be included in profile searches. For example, you might not want the ratings for competencies given by peers to display when other workers are searching person profiles.

**Default Instance Qualifier for Employee and Manager**

You can specify the default instance qualifier to use when managers and employees update a competency. Each time an employee or manager updates a competency, the record is assigned the instance qualifier code that is identified as the employee or manager default code.

**FAQs for Define Talent Profiles**

**What’s the difference between content sections and content subsections?**

Content subsections are for content types that have a parent content type. You add subsections to the content section for the parent content type.

Content sections are for content types that do not have a parent content type.

**Define Security for Customer Relationship Management**

**Security Tasks: Highlights**

Security tasks include the following.

- Security setup
- Security implementation and administration

**Note**
Perform security tasks in the integrated Oracle Fusion Applications user interfaces that are provided by the following products.

- Oracle Identity Manager (OIM)
- Oracle Authorization Policy Manager (APM)
- Oracle Fusion Human Capital Management (HCM) products
- Oracle Application Access Control Governor (AACG) in Governance, Risk, and Compliance Controls (GRCC)

Security administrative tasks performed by product administrators and implementation consultants, such as managing HCM security profiles, are presented in the documentation for those products.

**Prerequisite Tasks for Security Administration**

Sign into Oracle Fusion Applications for the first time with the super user account to create an IT security manager user account and provision it with the IT Security Manager role. With that account, perform the required security administration tasks.

- The super user account is established during installation. Refer to the Oracle Fusion Applications Installation Guide.
  
  See: Oracle Identity and Policy Management Configuration Parameters
- Create a user account and provision it with roles. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.
  
  See: Managing Users
- As a security guideline, provision a dedicated security professional with the IT Security Manager role as soon as possible after initial security setup and revoke that role from users provisioned with the Application Implementation Consultant role. If entitled to do so, see Security Tasks and Oracle Fusion Applications: How They Fit Together for details about provisioning the IT security manager.

**Required Security Administration Tasks**

Establish at least one implementation user and provision that user with sufficient access to set up the enterprise for all integrated Oracle Fusion Middleware and all application pillars or partitions. Initial security administration includes creating an implementation user, creating a data role for setting up HCM, and provisioning the implementation user with that data role so that the enterprise can be set up with Human Resources (HR) structures. Setup of business units, accounting structures, reference data sets, and so on does not require creation of an HCM administrator data role.

- Perform the initial security tasks. If entitled to do so, see Initial Security Administration: Critical Choices.
  
  See: User Management Tasks
- Create an implementation user. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.
  
  You can generate a data role using data role templates or HCM security profiles.
• For an overview of security tasks from the perspective of an applications administrator, refer to the Oracle Fusion Applications Administrator’s Guide

See: Understanding How to Secure Oracle Fusion Applications

Optional Security Administration Tasks

Once initial security administration is complete and your enterprise is set up with structures such as business units, additional security administration tasks are optional and based on modifying and expanding the predefined security reference implementation to fit your enterprise. See points to consider for defining security, data security and trading partner security after enterprise setup.

• Create users. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.

See: Creating Users

• Provision users with roles. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.

See: Adding and Removing Roles

• You manage users and job roles, including data and abstract roles, in Oracle Identity Management user interface pages. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

See: Interfaces


See: Managing Oracle Fusion Applications Data Security Policies

• You manage role provisioning rules in Human Capital Management (HCM). If entitled to do so, see Role Mappings: Explained.

• For a complete description of the Oracle Fusion Applications security reference implementation, see the Oracle Fusion Applications Security Reference Manuals for each offering.

• For a detailed functional explanation of the Oracle Fusion Applications security approach, refer to the following guides.

See: Oracle Fusion Applications Security Guide

See: Oracle Fusion Applications Security Hardening and Best Practices Guide

• Since security in Oracle Fusion Applications is based on integrations with Oracle Identity Management in Fusion Middleware, security features in the database, and Governance, Risk, Compliance, and Controls, additional resources in support of performing security tasks include the following.

• Authorization Policy Manager (APM) is available in Oracle Fusion Applications through integration with Oracle Identity Management
Authorization policy management involves managing duty roles, data role templates, and data security policies.


- Oracle Identity Management (OIM) is available in Oracle Fusion Applications through integration with Oracle Fusion Middleware. Identity management in Oracle Fusion Application involves creating and managing user identities, managing access control through user role assignment, managing enterprise roles, and managing workflow approvals and delegated administration.

See: Oracle Identity Management User's Guide

- Oracle Fusion Applications is certified to integrate with Applications Access Controls Governor (AACG) in the Oracle Governance, Risk and Compliance Controls (GRCC) suite to ensure effective SOD.

See: Oracle Application Access Controls Governor Users Guide
See: Oracle Application Access Controls Governor Implementation Guide

**Defining Security After Enterprise Setup: Points to Consider**

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Import Worker Users
- Import Partner Users
- Manage Job Roles
- Manage Duties
- Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human Capital Management (HCM), Supplier Portal, and Customer Relationship Management (CRM) Self Service, respectively, and associate them with users.

**Before Importing Users**

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Load
Batch Data task. If user and role provisioning rules have been defined, the Load Batch Data process automatically creates user and role provisioning requests as the workers are created.

Once the enterprise is set up, performing the Load Batch Data task populates the enterprise with HR workers in records linked by global user ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Load Batch Data task results in new user accounts being created. Worker email addresses as an alternate input for the Load Batch Data task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Load Batch Data task in HCM.

Note

The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in CRM.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

Import Users

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks.

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Load Batch Data task to create users from imported HR workers.

Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager.
In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

Manage Job Roles

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

Note

Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, CRM or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

Manage Duties

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering.

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.

In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.
Tip
As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

Note
Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

Manage Application Access Controls

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

Note
In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

Security Tasks and Oracle Fusion Applications: How They Fit Together

The major security tasks and their order within the context of an overall Oracle Fusion Applications implementation extend from security setup through production deployment audits.

The Oracle Fusion business process model (BPM) provides a sequence of security implementation tasks that includes the following.

- Security setup (Define Common Applications Configuration activity)
- Define Implementation Users task group (optional)
  - Create Implementation Users task
  - Create Data Role for Implementation Users task
• Assign Roles to Implementation Users task
• Define security - tasks vary depending on deployed Oracle Fusion product family
  • Revoke Data Role from Implementation Users task
  • Import Worker Users task
  • Import Partner Users task
  • Manage Duties task
  • Manage Job Roles task
  • Manage Application Access Controls task
• Define Automated Governance, Risk, and Performance Controls activity
  • Manage Application Access Controls task (AACG settings)
  • Manage Application Preventive Controls task
  • Manage Application Transaction Controls task
  • Manage Application Configuration Controls task
• User and role provisioning tasks
  • Implement Role Request and Provisioning Controls activity
    • Import Worker Users task
    • Import Partner Users task
    • Self Request User Roles task
    • Approve User and Role Provisioning Requests task
    • Assign User Roles task
    • Manage Supplier User Roles and User Role Usages task
  • Map and Synchronize User Account Details task
    • Tasks for viewing account details for self or others
    • Tasks for applying and managing various role provisioning rules
    • Tasks for running synchronization processes
• Security implementation and ongoing maintenance after setup (Manage IT Security activity)
  • Implement Function Security Controls
    • Create Job Role task
• Import Worker Users task
• Import Partner Users task
• Manage Duties task
• Manage Job Roles task
• Manage Users task
• Implement Data Security Controls
  • Manage Data Security Policies task
  • Manage Role Templates task
  • Manage Encryption Keys task
  • Manage Segment Security task
  • Manage Data Access Sets task
  • Define Security Profiles task group
• Auditing tasks
  • Manage Security Audit, Compliance and Reporting activity
  • Manage Application Access Controls task

**Note**
Go live deployment does not require lockdown or specific security tasks because security is enforced across the test to production information life cycle.

**Required Roles**
The Oracle Fusion Applications installation process sets up a single, super user provisioned with the following enterprise roles.

- Application Implementation Consultant
- IT Security Manager
- Application Administrators for the provisioned products

Additionally, provision the super user with the following roles to enable user and role administration in Oracle Identity Management (OIM) and the Authorization Policy Manager (APM).

- Identity User Administrator
- Role Administrator
- APM Administration Application

**Important**
As a security guideline, provision a dedicated security professional with the IT Security Manager role as soon as possible after initial security setup and revoke that role from users provisioned with the Application Implementation Consultant role.
The user provisioned with the IT Security Manager role should additionally be provisioned with the following roles.

- Identity User Administrator for access to user administration in OIM
- Role Administrator for access to role administration in OIM
- APM Administration Application role for access to administration in the Authorization Policy Manager

Provision these roles to the IT security manager using the XELSYSADM user in OIM.

**Tools Used to Perform Security Tasks**

Security tasks are supported by tools within both Oracle Fusion Applications and Oracle Fusion Middleware.

The figure lists the tasks associated with each of the integrated products and pillars of an Oracle Fusion Applications deployment.
### Security Tasks: Overview

Security tasks span multiple business processes and are performed by various roles using numerous integrated tools.

The following table shows the business process model (BPM) tasks and tools used to support securing Oracle Fusion Applications.

<table>
<thead>
<tr>
<th>Example Task</th>
<th>Oracle BPM Task</th>
<th>Supporting Tools</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>View duty roles inherited by a job role</td>
<td>Manage Duties</td>
<td>• Authorization Policy Manager (APM)</td>
<td>Each logical partition or pillar contains a collection of application roles, and function and data security policies.</td>
</tr>
<tr>
<td>View entitlement or policies carried by a job role</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>In LDAP, the policy store stores application roles and the identity store stores enterprise roles.</td>
</tr>
<tr>
<td>Add a job role to a role hierarchy</td>
<td>Manage Job Roles</td>
<td>• Oracle Identity Management (OIM)</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Add a duty role to a role hierarchy</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>LDAP stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions.</td>
</tr>
<tr>
<td>Create a hierarchy of enterprise (abstract, job, data) roles</td>
<td>Manage Job Roles</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td>Create a hierarchy of (application) duty roles</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Create a new job role</td>
<td>1. Create Job Roles 2. Manage Job Roles</td>
<td>• OIM</td>
<td>The policy store stores duty roles. The identity store stores enterprise roles. Some duty roles may enable actions and their associated users interface features that your enterprise does not want users to perform in Oracle Fusion applications.</td>
</tr>
<tr>
<td>Change duty roles inherited by a job or abstract role</td>
<td>Manage Duties</td>
<td>• APM</td>
<td></td>
</tr>
<tr>
<td>Create a new duty role</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>All functions and actions in Oracle Fusion Applications that need to be secured are covered by the reference implementation. In some cases, especially with function customizations, a new duty role may be needed.</td>
</tr>
<tr>
<td>View Segregation of Duties (SOD) policies respected by a duty role</td>
<td>Manage Application Access Controls</td>
<td>• Application Access Controls Governor (AACG) in Governance, Risk, and Compliance Controls (GRCC)</td>
<td>The Security Reference Manuals (SRM) document the segregation of duties (SOD) policies respected within each job role</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>View SOD policy violations carried by the duty roles inherited by a job role</td>
<td>Manage Application Access Controls</td>
<td>• AACG in GRCC</td>
<td>The Security Reference Manuals (SRM) document the SOD policies respected within each job role</td>
</tr>
<tr>
<td>View SOD policy violations</td>
<td>Manage Segregation of Duties Policies</td>
<td>• AACG in GRC</td>
<td>The SRM documents the SOD conflicts for each job role</td>
</tr>
<tr>
<td>View the data security policies carried by a job, abstract, and data roles</td>
<td>Manage Data Security Policies</td>
<td>• APM</td>
<td>Oracle Fusion Data Security stores data security policies in the policy store. Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>Create and update HCM security profiles</td>
<td>Manage Data Role and Security Profiles</td>
<td>• Oracle Fusion HCM</td>
<td>This task does not include assigning data roles to the users, which is supported by user provisioning tasks.</td>
</tr>
<tr>
<td>Create (generate) a data role</td>
<td>1. Manage Role Templates 2. Manage Data Roles and Security Profiles</td>
<td>• APM 2. Oracle Fusion HCM</td>
<td>Data roles are generated automatically based on data role templates and enterprise setup. Changes to data role templates generate new or changed data roles. Create data roles in HCM using the Manage Data Roles and Security Profiles task.</td>
</tr>
<tr>
<td>Create a new data security policy (not through generated data roles based on data role templates or HCM security profiles)</td>
<td>Manage Data Security Policies</td>
<td>• APM</td>
<td>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>View data role templates defined by a product</td>
<td>Manage Role Templates</td>
<td>• APM</td>
<td></td>
</tr>
<tr>
<td>Create or edit an existing data role template</td>
<td>Manage Role Templates</td>
<td>• APM</td>
<td></td>
</tr>
<tr>
<td>Secure common objects such as attachment categories or profile options</td>
<td>Manage Data Security Policies</td>
<td>• APM</td>
<td>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>View, create, update encryption keys used to secure attributes of personally identifiable information</td>
<td>Manage Encryption Keys</td>
<td>• Oracle Fusion Payments</td>
<td></td>
</tr>
<tr>
<td>View, create, update Data Access Sets used to secure Ledgers and Ledger Sets</td>
<td>Manage Data Access Sets</td>
<td>• Oracle Fusion General Ledger</td>
<td></td>
</tr>
<tr>
<td>View, create, update accounting flexfield segment security rules</td>
<td>Manage Security Segments</td>
<td>• Oracle Fusion General Ledger</td>
<td></td>
</tr>
<tr>
<td>View or update the set of job roles that can be provisioned to supplier users</td>
<td>Manage Supplier User Role</td>
<td>• Supplier Portal • Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Determine the supplier job roles that the supplier self service administrator can provision to supplier users</td>
<td>Manage Supplier User Role Usages</td>
<td>• Supplier Portal • Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Set default supplier job roles based on the set of supplier roles that are defined by performing the Manage Supplier User Roles task</td>
<td>Manage Supplier User Role Usages</td>
<td>• Supplier Portal • Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Create a new implementation user</td>
<td>Create Implementation Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td>Import legacy users</td>
<td>• Import Worker Users • Import Partner Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td>Create a new user</td>
<td>Manage Users</td>
<td>• HCM</td>
<td>HCM creates a new user and identity when a new worker is created. The Hire Employee and Add Contingent Worker tasks also result in new user creation requests. Creating a new user automatically triggers role provisioning requests based on role provisioning rules.</td>
</tr>
<tr>
<td>Task</td>
<td>Steps</td>
<td>Tools</td>
<td>Notes</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
</tbody>
</table>
| Provision roles to a user | 1. Provision Roles to Implementation Users  
2. Manage Users | - OIM  
- Oracle Fusion HCM  
- Oracle Fusion CRM  
- Oracle Fusion Suppliers | Implementation users are provisioned through OIM since HCM is not setup at the start of the implementation. The Provision Roles to Implementation Users is not needed once implementation is complete. Once HCM is setup, HCM is used to provision roles to non-implementation users by performing the Manage Users task. Human Resources (HR) transaction flows such as Hire and Promote also provision roles. Once supplier users are setup, Supplier Model can be used by internal users to maintain supplier user accounts or supplier users can maintain their accounts in Supplier Portal. |
| View the job, abstract, and data roles provisioned to a user | 1. Manage Users  
2. Manage User Principal  
3. Provision Roles to Implementation Users | - Human Capital Management (HCM)  
- OIM | LDAP stores users, roles and provisioning information. The Manage User Principal and Provision Roles to Implementation Users tasks are not needed once implementation is complete. |
| Revoke role from user | Manage Users | - HCM | You can revoke roles from various Human Resources task flows, the HCM Manage Users task and OIM. User termination includes role revocation. |
| Approve role provisioning or user account request | Approve User and Role Provisioning Requests | - OIM | |
| View audit logs | Not applicable | - Oracle Enterprise Manager | Viewing audit logs is a Oracle Fusion Middleware function and not represented by an Oracle Fusion Applications BPM task. |
Define Data Security for Customer Relationship Management

Data Security: Explained

By default, users are denied access to all data.

Data security makes data available to users by the following means.

- Policies that define grants available through provisioned roles
- Policies defined in application code

You secure data by provisioning roles that provide the necessary access. Enterprise roles provide access to data through data security policies defined for the inherited application roles.

When setting up the enterprise with structures such as business units, data roles are automatically generated that inherit job roles based on data role templates. Data roles also can be generated based on HCM security profiles. Data role templates and HCM security profiles enable defining the instance sets specified in data security policies.

When you provision a job role to a user, the job role implicitly limits data access based on the data security policies of the inherited duty roles. When you provision a data role to a user, the data role explicitly limits the data access of the inherited job role to a dimension of data.

Data security consists of privileges conditionally granted to a role and used to control access to the data. A privilege is a single, real world action on a single business object. A data security policy is a grant of a set of privileges to a principal on an object or attribute group for a given condition. A grant authorizes a role, the grantee, to actions on a set of database resources. A database resource is an object, object instance, or object instance set. An entitlement is one or more allowable actions applied to a set of database resources.

Data is secured by the following means.

<table>
<thead>
<tr>
<th>Data security feature</th>
<th>Does what?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data security policy</td>
<td>Grants access to roles by means of entitlement</td>
</tr>
<tr>
<td>Role</td>
<td>Applies data security policies with conditions to users through role provisioning.</td>
</tr>
<tr>
<td>Data role template</td>
<td>Defines the data roles generated based on enterprise setup of data dimensions such as business unit.</td>
</tr>
<tr>
<td>HCM security profile</td>
<td>Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Masking</td>
<td>Hides private data on non-production database instances</td>
</tr>
<tr>
<td>Encryption</td>
<td>Scrambles data to prevent users without decryption authorization from reading secured data</td>
</tr>
</tbody>
</table>

The sets of data that a user can access via roles are defined in Oracle Fusion Data Security. Oracle Fusion Data Security integrates with Oracle Platform Security Services (OPSS) to entitle users or roles (which are stored externally) with access to data. Users are granted access through the entitlement assigned to the roles or role hierarchy with which the user is provisioned. Conditions are WHERE clauses that specify access within a particular dimension, such as by business unit to which the user is authorized.

### Data Security Policies

Data security policies articulate the security requirement "Who can do What on Which set of data," where 'Which set of data' is an entire object or an object instance or object instance set and 'What' is the object entitlement.

For example, accounts payable managers can view AP disbursements for their business unit.

<table>
<thead>
<tr>
<th>Who</th>
<th>can do</th>
<th>what</th>
<th>on which set of data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounts payable managers</td>
<td>view</td>
<td>AP disbursements</td>
<td>for their business unit</td>
</tr>
</tbody>
</table>

A data security policy is a statement in a natural language, such as English, that typically defines the grant by which a role secures business objects. The grant records the following.

- Table or view
- Entitlement (actions expressed by privileges)
- Instance set (data identified by the condition)

For example, disbursement is a business object that an accounts payable manager can manage by payment function for any employee expenses in the payment process.

#### Note

Some data security policies are not defined as grants but directly in applications code. The security reference manuals for Oracle Fusion Applications offerings differentiate between data security policies that define a grant and data security policies defined in Oracle Fusion applications code.

A business object participating in a data security policy is the database resource of the policy.
Data security policies that use job or duty roles refer to data security entitlement.

For example, the data security policy for the Accounts Payable Manager job role refers to the view action on AP disbursements as the data security entitlement.

**Important**

The duty roles inherited by the job role can be moved and job roles reassembled without having to modify the data security.

As a security guideline, data security policies based on user session context should entitle a duty role. This keeps both function and data security policies at the duty role level, thus reducing errors.

For example, a Sales Party Management Duty can update Sales Party where the provisioned user is a member of the territory associated with the sales account. Or the Sales Party Management Duty can update Sales Party where the provisioned user is in the management chain of a resource who is on the sales account team with edit access. Or the Participant Interaction Management Duty can view an Interaction where the provisioned user is a participant of the Interaction.

For example, the Disbursement Process Management Duty role includes entitlement to build documents payable into payments. The Accounts Payable Manager job role inherits the Disbursement Process Management Duty role. Data security policies for the Disbursement Process Management Duty role authorize access to data associated with business objects such as AP disbursements within a business unit. As a result, the user provisioned with the Accounts Payable Manager job role is authorized to view AP disbursements within their business unit.

A data security policy identifies the entitlement (the actions that can be made on logical business objects or dashboards), the roles that can perform those actions, and the conditions that limit access. Conditions are readable WHERE clauses. The WHERE clause is defined in the data as an instance set and this is then referenced on a grant that also records the table name and required entitlement.

**Data Roles**

Data roles are implemented as job roles for a defined set of data.

A data role defines a dimension of data within which a job is performed. The data role inherits the job role that describes the job. For example, a data role entitles a user to perform a job in a business unit.

The data role inherits abstract or job roles and is granted data security privileges. Data roles carry the function security privileges inherited from job roles and also the data security privilege granted on database objects and table rows.

For example, an accounts payables specialist in the US Business Unit may be assigned the data role Accounts Payables Specialist - US Business Unit. This data role inherits the job role Accounts Payables Specialist and grants access to transactions in the US Business Unit.

A data role may be granted entitlement over a set people.
For example, a Benefits Administrator A-E is allowed to administer benefits for all people that have a surname that begins with A-E.

Data roles are created using data role templates. You create and maintain data roles in the Authorization Policy Manager (APM). Use the Manage Data Roles and Security Profiles task to create and maintain HCM data roles in Oracle Fusion HCM.

**HCM Security Profiles**

HCM security profiles are used to secure HCM data, such as people and departments. You use HCM security profiles to generate grants for an enterprise role. The resulting data role with its role hierarchy and grants operates in the same way as any other data role.

For example, an HCM security profile identifies all employees in the Finance division.

Oracle Fusion Payroll uses HCM security profiles to secure project organizations. Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

**Masking and Encryption**

Oracle Fusion Applications uses masking to protect sensitive data from view by unauthorized users. Encryption APIs mask sensitive fields in applications user interfaces. Additionally, Oracle Data Masking is available for masking data in non-production instances and Oracle Transparent Data Encryption is available for protecting data in transit or in backups independent of managing encryption keys.

**Defining Data Security After Enterprise Setup: Points to Consider**

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Data Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Manage Data Access Sets
- Manage Segment Security
- Manage Role Templates
- Manage Data Security Policies
- Manage Encryption Keys

These tasks address data security administration. For information on using the user interface pages for setting up and managing data security, see the Oracle

**Note**

The Manage Data Role and Security Profiles task, and all other HCM security profile setup tasks are documented in Human Capital Management (HCM) documentation.

**Manage Data Access Sets**

Data access sets define a set of access privileges to one or more ledgers or ledger sets.

The information on ledgers that are attached to data access sets are secured by function security. Users must have access to the segment values associated with the data access sets to access the corresponding GL account.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Data Access Administration Duty role, which is entitled to manage data access sets (the entitlement is Define General Ledger Data Access Set). This entitlement provides the access necessary to perform the Manage Data Access Sets task in General Ledger.

**Manage Segment Security**

Balancing or management segment values can secure data within a ledger.

Segment values are stored in GL_ACCESS_SET_ASSIGNMENTS and secured by restrictions, such as Exclude, on parameters that control the set of values that a user can use during data entry.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Key Flexfield Administration Duty role, which is entitled to manage application key flexfields (the entitlement is Manage Application Key Flexfield). This entitlement provides the access necessary to perform the Manage Segment Security task in General Ledger.

**Manage Role Templates**

Data role templates automatically create or update data roles based on dimensions such as business unit. As an enterprise expands, data role templates trigger replication of roles for added dimensions. For example, when creating a new business unit, a data role template generates a new Accounts Payables Manager data role based on the Financials Common Module Template for Business Unit Security data role template.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage data role templates (the entitlement is Manage Role Template). This entitlement provides the access necessary to perform the Manage Role Templates task in APM.
Manage Data Security Policies

Data security grants provisioned to roles are data security policies. The security reference implementation provides a comprehensive set of predefined data security policies and predetermined data security policies based on data role templates.

Data security policies are available for review in Authorization Policy Manager (APM). Data security policies are implemented by grants stored in Oracle Fusion Data Security (FND_GRANTS).

Data security policies secure the database resources of an enterprise. Database resources are predefined applications data objects and should not be changed. However, for cases where custom database resources must be secured objects, the IT security manager is entitled to manage database resources and create new data security policies.

Warning

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage data security policies (the entitlement is Manage Data Security Policy). This entitlement provides the access necessary to perform the Manage Data Security Policies task in APM.

Manage Encryption Keys

Create or edit encryption keys held in Oracle Wallet to secure Personally Identifiable Information (PII) attributes This task is only available when Payments is implemented.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Payments Data Security Administration Duty role, which is entitled to manage encryption keys that secure PII (the entitlement is Manage Wallet). This entitlement provides the access necessary to perform the Manage Encryptions Keys task in Payments.

Data Security in the Security Reference Implementation: Explained

The reference implementation contains a set of data security policies that can be inspected and confirmed to be suitable or a basis for further implementation using the Authorization Policy Manager (APM).

The security implementation of an enterprise is likely a subset of the reference implementation, with the enterprise specifics of duty roles, data security policies, and HCM security profiles provided by the enterprise.
The business objects registered as secure in the reference implementation are database tables and views.

Granting or revoking object entitlement to a particular user or group of users on an object instance or set of instances extends the base Oracle Fusion Applications security reference implementation without requiring customization of the applications that access the data.

**Data Security Policies in the Security Reference Implementation**

The data security policies in the reference implementation entitle the grantee (a role) to access instance sets of data based on SQL predicates in a WHERE clause.

**Tip**

When extending the reference implementation with additional data security policies, identify instance sets of data representing the business objects that need to be secured, rather than specific instances or all instances of the business objects.

Predefined data security policies are stored in the data security policy store, managed in the Authorization Policy Manager (APM), and described in the Oracle Fusion Applications Security Reference Manual for each offering. A data security policy for a duty role describes an entitlement granted to any job role that includes that duty role.

**Warning**

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

The reference implementation only enforces a portion of the data security policies in business intelligence that is considered most critical to risk management without negatively affecting performance. For performance reasons it is not practical to secure every level in every dimension. Your enterprise may have a different risk tolerance than assumed by the security reference implementation.

**HCM Security Profiles in the Security Reference Implementation**

The security reference implementation includes some predefined HCM security profiles for initial usability. For example, a predefined HCM security profile allows line managers to see the people that report to them.

The IT security manager uses HCM security profiles to define the sets of HCM data that can be accessed by the roles that are provisioned to users.

**Data Roles**

The security reference implementation includes no predefined data roles to ensure a fully secured initial Oracle Fusion Applications environment.
The security reference implementation includes data role templates that you can use to generate a set of data roles with entitlement to perform predefined business functions within data dimensions such as business unit. Oracle Fusion Payables invoicing and expense management are examples of predefined business functions. Accounts Payable Manager - US is a data role you might generate from a predefined data role template for payables invoicing if you set up a business unit called US.

HCM provides a mechanism for generating HCM related data roles.

**Securing Data Access: Points to Consider**

Oracle Fusion Applications supports securing data through role-based access control (RBAC) by the following methods.

<table>
<thead>
<tr>
<th>Method of securing data</th>
<th>Reason</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data roles apply explicit data security policies on job and abstract roles</td>
<td>Appropriate for job and abstract roles that should only access a subset of data, as defined by the data role template that generates the data role or by HCM security profiles.</td>
<td>Accounts Payable Manager - US data role to provide an accounts payable manager in the US business unit with access to invoices in the US business unit.</td>
</tr>
<tr>
<td>Data security policies</td>
<td>Define data access for application roles and provide inheriting job and abstract roles with implicit data security</td>
<td>Projects</td>
</tr>
</tbody>
</table>

If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

When a runtime session is created, Oracle Platform Security Services (OPSS) propagates only the necessary user to role mapping based on Oracle Fusion Data Security grants. A grant can specify entitlement to the following.

- Specific rows of data (data object) identified by primary key
- Groups of data (instance set) based on a predicate that names a particular parameter
- Data objects or instance sets based on runtime user session variables

Data is either identified by the primary key value of the row in the table where the data is stored. Or data is identified by a rule (SQL predicate) applied to the WHERE clause of a query against the table where the data is stored.

**Grants**

Oracle Fusion Data Security can be used to restrict the following.

- Rows that are returned by a given query based on the intended business operation
- Actions that are available for a given row
Grants control which data a user can access.

Note

Attribute level security using grants requires a data security policy to secure the attribute and the entitlement check enforces that policy.

A grant logically joins a user or role and an entitlement with a static or parameterized object instance set. For example, `REGION='WEST'` is a static object instance set and `REGION=\&GRANT_ALIAS.PARAMETER1` is a parameterized object instance set. In the context of a specific object instance, grants specify the allowable actions on the set of accessible object instances. In the database, grants are stored in `FND_GRANTS` and object instance sets are stored in `FND_OBJECT_INSTANCE_SETS`. Object access can be tested using the privilege check application programming interface (API).

Securing a Business Object

A business object is a logical entity that is typically implemented as a table or view, and corresponds to a physical database resource. The data security policies of the security reference implementation secure predefined database resources. Use the Manage Data Security Policies task to define and register other database resources.

Data security policies identify sets of data on the registered business object and the actions that may be performed on the business object by a role. The grant can be made by data instance, instance set or at a global level.

Note

Use parameterized object instance sets whenever feasible to reduce the number of predicates the database parses and the number of administrative intervention required as static object instances sets become obsolete. In HCM, security profiles generate the instance sets.

Database Resources and Data Security Policies: How They Work Together

A data security policy applies a condition and allowable actions to a database resource for a role. When that role is provisioned to a user, the user has access to data defined by the policy. In the case of the predefined security reference implementation, this role is always a duty role. Data roles generated to inherit the job role based on data role templates limit access to database resources in a particular dimension, such as the US business unit.

The database resource defines and instance of a data object. The data object is a table, view, or flexfield.

The following figure shows the database resource definition as the means by which a data security policy secures a data object. The database resource names the data object. The data security policy grants to a role access to that database resource based on the policy’s action and condition.
Database Resources
A database resource specifies access to a table, view, or flexfield that is secured by a data security policy.

- Name providing a means of identifying the database resource
- Data object to which the database resource points

Data Security Policies
Data security policies consist of actions and conditions for accessing all, some, or a single row of a database resource.

- Condition identifying the instance set of values in the data object
- Action specifying the type of access allowed on the available values

Note
If the data security policy needs to be less restrictive than any available database resource for a data object, define a new data security policy.

Actions
Actions correspond to privileges that entitle kinds of access to objects, such as view, edit, or delete. The actions allowed by a data security policy include all or a subset of the actions that exist for the database resource.
Conditions

A condition is either a SQL predicate or an XML filter. A condition expresses the values in the data object by a search operator or a relationship in a tree hierarchy. A SQL predicate, unlike an XML filter, is entered in a text field in the data security user interface pages and supports more complex filtering than an XML filter, such as nesting of conditions or sub queries. An XML filter, unlike a SQL predicate, is assembled from choices in the UI pages as an AND statement.

Tip

An XML filter can be effective in downstream processes such as business intelligence metrics. A SQL predicate cannot be used in downstream metrics.

Data Role Templates: Explained

You use data role templates to generate data roles. You generate such data roles, and create and maintain data role templates in the Authorization Policy Manager (APM).

Note

HCM data roles are generated using the Manage Data Roles and Security Profiles task, which uses HCM security profiles, not data role templates, to define the data security condition.

The following attributes define a data role template.

- Template name
- Template description
- Template group ID
- Base roles
- Data dimension
- Data role naming rule
- Data security policies

The data role template specifies which base roles to combine with which dimension values for a set of data security policies. The base roles are the parent job or abstract roles of the data roles.

Note

Abstract, job, and data roles are enterprise roles in Oracle Fusion Applications. Oracle Fusion Middleware products such as Oracle Identity Manager (OIM) and Authorization Policy Manager (APM) refer to enterprise roles as external roles. Duty roles are implemented as application roles in APM and scoped to individual Oracle Fusion Applications.

The dimension expresses stripes of data, such as territorial or geographic information you use to partition enterprise data. For example, business units are a type of dimension, and the values picked up for that dimension by the
data role template as it creates data roles are the business units defined for your enterprise. The data role template constrains the generated data roles with grants of entitlement to access specific data resources with particular actions. The data role provides provisioned users with access to a dimensional subset of the data granted by a data security policy.

An example of a dimension is a business unit. An example of a dimension value is a specific business unit defined in your enterprise, such as US. An example of a data security policy is a grant to access a business object such as an invoice with a view entitlement.

When you generate data roles, the template applies the values of the dimension and participant data security policies to the group of base roles.

The template generates the data roles using a naming convention specified by the template’s naming rule. The generated data roles are stored in the Lightweight Directory Access Protocol (LDAP) store. Once a data role is generated, you provision it to users. A user provisioned with a data role is granted permission to access the data defined by the dimension and data security grant policies of the data role template.

For example, a data role template contains an Accounts Payable Specialist role and an Accounts Payable Manager role as its base roles, and region as its dimension, with the dimension values US and UK. The naming convention is [base-role-name]:[DIMENSION-CODE-NAME]. This data role template generates four data roles.

- Accounts Payable Specialist - US (business unit)
- Accounts Payable Specialist - UK (business unit)
- Accounts Payable Manager - US (business unit)
- Accounts Payable Manager - UK (business unit)

Making Changes To Data Role Templates

If you add a base role to an existing data role template, you can generate a new set of data roles. If the naming rule is unchanged, existing data roles are overwritten.

If you remove a base role from a data role template and regenerate data roles, a resulting invalid role list gives you the option to delete or disable the data roles that would be changed by that removal.

Making Changes to Dimension Values

If you add a dimension value to your enterprise that is used by a data role template, you must regenerate roles from that data role template to create a data role for the new dimension. For example if you add a business unit to your enterprise, you must regenerate data roles from the data role templates that include business unit as a dimension.

If you add or remove a dimension value from your enterprise that is used to generate data roles, regenerating the set of data roles adds or removes the data roles for those dimension values. If your enterprise has scheduled regeneration as an Oracle Enterprise Scheduler Services process, the changes are made automatically.
For information on working with data role templates, see the Oracle Fusion Middleware Administrator’s Guide for Authorization Policy Manager (Oracle Fusion Applications Edition).

HCM Data Roles: Explained

HCM data roles, like all Oracle Fusion Applications data roles, define data security policies: they enable users to perform a set of tasks, using identified menus, menu items, and pages in application user interfaces, on a specified set of data within those user interfaces. Because data roles are specific to the enterprise, no predefined HCM data roles exist.

How HCM Data Roles Differ from Other Data Roles

HCM data roles differ from other data roles in the following ways:

- You create and maintain HCM data roles outside the Oracle Fusion Middleware Authorization Policy Manager (APM), and they are not based on data role templates.
  
  Although HCM data roles are visible in the Oracle Fusion Middleware APM, they must not be maintained there.

- A single HCM data role can enable access to data of multiple types.
  
  You identify the data that users can access in HCM security profiles. You can create security profiles for the person, organization, position, country, legislative data group (LDG), document type, payroll, and payroll flow objects.

Selecting the Job Role

Each HCM data role is associated with a single job role, which you select from the list of enterprise roles. The HCM securing objects that the selected role needs to access are identified automatically, and the appropriate types of security profile are displayed. For example, if you select the job role human resource analyst, users with that job role need to access managed person, public person, organization, position, LDG, and document type data; therefore, security profiles for those object types must be included in the HCM data role. The security profile types that appear in the HCM data role vary according to the data requirements of the selected job role.

If you select a job role that requires no access to HCM data secured by security profiles, you cannot create an HCM data role.

Creating or Selecting the Security Profiles

You can either create new security profiles or use existing security profiles. For each object type, you can include only one security profile in an HCM data role.

Users with Multiple HCM Data Roles

When users have multiple HCM data roles, the data security policies arising from each role remain separate. For example, being able to promote or terminate
workers in the purchasing department in one HCM data role and view contact
details of all workers in the sales department in another HCM data role does not
enable a user to promote or terminate workers in the sales department.

Components of the HCM Data Role

The following figure summarizes how the components of the HCM data role
contribute to Oracle Fusion Data Security for the data role. Oracle Fusion Data
Security comprises the data security policies for data roles that are generated
automatically when data roles are created.

The job role that you select in the HCM data role inherits multiple duty roles.
Each duty role has one or more function privileges and related data privileges,
from which the relevant HCM objects are identified. The specific instances of
the objects required by this HCM data role are identified in security profiles and
stored in a data instance set. Data security policy data is created automatically in
Oracle Fusion Data Security when you create the data role.

For example, the human resource specialist job role inherits the employee hire
and worker promotion duty roles, among many others. The inherited duty roles
provide both function privileges, such as Hire Employee, Rehire Employee,
and Promote Workers, and data privileges to HCM objects, such as person and assignment. The specific instances of those objects required by this HCM data role, such as people with assignments in a specified legal employer and department, are identified in security profiles.

**HCM Security Profiles: Explained**

A security profile defines the criteria that identify instances of a human capital management (HCM) object. For example, a person security profile defines the criteria that identify one or more person records, and a position security profile defines the criteria that identify one or more positions. When you include a security profile in an HCM data role and provision the data role to a user, that user can access the data instances identified in the security profile. The type of access available to the user (for example whether the user can edit or simply view the data) depends on the job role identified in the HCM data role.

**HCM Object Types**

You can create security profiles for the following HCM object types:

- Person
  - Managed person
  - Public person
- Organization
- Position
- Legislative data group (LDG)
- Country
- Document type
- Payroll
- Payroll flow

All security profile definitions for these HCM objects are eventually visible in the Oracle Fusion Middleware Authorization Policy Manager (APM). The name of the security profile’s data instance set in the Oracle Fusion Middleware APM is derived from the name of the security profile and the relevant object type. For example, if the security profile name is Manager Hierarchy, then the data instance set for the object PER_ALL_PEOPLE_F is HCM:PER:PER_ALL_PEOPLE_F:Manager Hierarchy.

You must use the Oracle Fusion Human Capital Management interfaces, which are designed for ease of use and access, to create and maintain security profiles; do not use the Oracle Fusion Middleware APM to maintain security profiles for these HCM objects.

**Security Criteria in HCM Security Profiles**

In any HCM security profile, you specify the criteria that identify data instances of the relevant type. For example, in an organization security profile, you can identify organizations by organization hierarchy, by organization classification, or by listing organizations to include in or exclude from the security profile.
All of the criteria in an HCM security profile apply when the data instance set is defined; for example, if you identify organizations by both organization hierarchy and organization classification, then both sets of criteria apply, and only those organizations that satisfy all criteria belong to the data instance set.

**Predefined HCM Security Profiles**

The following HCM security profiles are predefined:

<table>
<thead>
<tr>
<th>Security Profile Name</th>
<th>HCM Security Profile Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View All People</td>
<td>Person</td>
<td>Identifies all person records in the enterprise</td>
</tr>
<tr>
<td>View Own Record</td>
<td>Person</td>
<td>Identifies the signed-on user's own person record and the person records of that user's contacts</td>
</tr>
<tr>
<td>View Manager Hierarchy</td>
<td>Person</td>
<td>Identifies the signed-on manager's hierarchy</td>
</tr>
<tr>
<td>View All Workers</td>
<td>Person</td>
<td>Identifies the person records of all people who have a work relationship in the enterprise</td>
</tr>
<tr>
<td>View All Organizations</td>
<td>Organization</td>
<td>Identifies all organizations in the enterprise</td>
</tr>
<tr>
<td>View All Positions</td>
<td>Position</td>
<td>Identifies all positions in the enterprise</td>
</tr>
<tr>
<td>View All Legislative Data Groups</td>
<td>LDG</td>
<td>Identifies all LDGs in the enterprise</td>
</tr>
<tr>
<td>View All Countries</td>
<td>Country</td>
<td>Identifies all countries in the FND_TERRITORIES table</td>
</tr>
<tr>
<td>View All Document Types</td>
<td>Document Type</td>
<td>Identifies all document types in the enterprise</td>
</tr>
<tr>
<td>View All Payrolls</td>
<td>Payroll</td>
<td>Identifies all payrolls in the enterprise</td>
</tr>
<tr>
<td>View All Payroll Flows</td>
<td>Payroll Flow</td>
<td>Identifies all payroll flows in the enterprise</td>
</tr>
</tbody>
</table>

You can include the predefined security profiles in any HCM data role, but you cannot edit them. Note also that the View all option is disabled in any security profile that you create; this restriction exists because predefined security profiles exist for this requirement.

**Creating Security Profiles**

You can create security profiles either individually or as part of the process of creating an HCM data role. If you have standard requirements, it may be more efficient to create the security profiles individually and include them in appropriate HCM data roles.

**Reusability and Inheritance of Security Profiles**

Regardless of how you create them, all security profiles are reusable; they do not belong to particular HCM data roles, and you can include them in any HCM data role for which they define an appropriate data instance set.
You can include security profiles in other security profiles. For example, you can include an organization security profile:

- In a person security profile, to secure person records by department, business unit, or legal employer
- In a position security profile, to secure positions by department or business unit

Therefore, one security profile can inherit the data instance set defined by another.

**Setting Up Data Security for Employees: Worked Example**

Oracle Fusion Applications users may need to access Oracle Fusion Human Capital Management (HCM) person data, such as lists of person names, in their product interfaces. To provide this access, you assign predefined HCM security profiles to relevant abstract roles, such as Employee. This example shows how to assign security profiles to the Employee abstract role.

**Searching for the Abstract Role**

1. In the Functional Setup Manager (FSM), click Go to Task for the Manage Data Role and Security Profiles task.
2. On the Manage HCM Data Roles page, enter the abstract role name Employee in the Role field.
3. Click Search.
4. In the search results, highlight the entry for the Employee role.
5. Click Assign.

**Assigning Security Profiles to the Abstract Role**

1. On the Assign Data Role: Security Criteria page use the default values, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position Security Profile</td>
<td>View All Positions</td>
</tr>
<tr>
<td>Country Security Profile</td>
<td>View All Countries</td>
</tr>
<tr>
<td>LDG Security Profile</td>
<td>View All Legislative Data Groups</td>
</tr>
<tr>
<td>Person Security Profile (Person section)</td>
<td>View Own Record</td>
</tr>
<tr>
<td>Person Security Profile (Public Person section)</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Document Type Security Profile</td>
<td>View All Document Types</td>
</tr>
</tbody>
</table>

These are the security profiles that are typically assigned to the Employee abstract role. You may see a subset of these security profiles, depending on the combination of product offerings that you are implementing.

2. Click Review.
Define Users for Customer Relationship Management

Securing Identities and Users: Points To Consider

Identity covers all aspects of an entity’s existence within the contexts in which it is used. The identity of an enterprise user consists of HR attributes, roles, resources, and relationships.

HR attributes include identifying information about a user that is relatively static and well understood, such as first and last name, title, and job function.

Roles are part of a user’s identity and define the user’s purpose and responsibilities.

Within identity management, resources define what a user can and does do. In an enterprise, this typically translates into what resources a user has access to, what privileges they have on that resource, and what they have been doing on that resource. Resources can be application accounts or physical devices such as laptops or access cards. The enterprise owns the resources, secures them, and manages access to the resources by managing the user’s identity and access.

Relationships establish the portion of user identities that involve organizational transactions such as approvals.

An Oracle Fusion Applications user and corresponding identity are usually created in a single transaction, such as when a worker is created in Human Resources (HR). That transaction automatically triggers provisioning requests for the user based on role provisioning rules.

User accounts for some identities that are not employees, such as partner contacts, may be created in a later transaction using an identity that is already created in the identity store. Supplier contacts are created in the Supplier Model, not HR.

Stores

Various locations store identity and user data.

Identity data consists of the following.

- HR person records
- Oracle Fusion Trading Community Model party records

In Oracle Fusion Applications, identities and users correspond one to one, but not all identities correspond to a user, and not all users are provisioned with an identity. Some identities stored in HR and Trading Community Model may not be provisioned to user accounts and therefore are not synchronized with Oracle Identity Management (OIM). For example, a contact for a prospective customer is an identity in Trading Community Model but may not be provisioned with a user account in OIM. Some users stored in the Lightweight Directory Access Protocol (LDAP) store may not be provisioned with identities. For example, system user accounts used to run Web services to integrate third party services with Oracle Fusion Applications are not associated with a person record in HR or Trading Community Model. Some identifying credentials such as name,
department, e-mail address, manager, and location are stored with user data in the LDAP store.

**Importing Users**

You can import users or user attributes in bulk from existing legacy identity and user stores.

Your tasks may include the following.

- Create users in bulk
- Update specific attributes for all users, such as postal code
- Link users to HR or Trading Community Model persons
- Monitor progress of the import process
- Correct errors & re-import
- Export users in bulk
- Import and export users using a standard plain text data interchange format like Lightweight Data Interchange Format (LDIF)

You can reserve a specific user name not currently in use for use in the future, or release a reserved username from the reservation list and make it available for use. Between a user registration request and approved registration, Oracle Fusion Applications holds the requested user name on the reservation list, and releases the name if an error occurs in the self-registration process or the request is rejected. Self-registration processes check the reservation list for user name availability and suggest alternative names.

**Provisioning Events**

New identities, such as new hires, trigger user and role provisioning events. In addition to user creation tasks, other tasks, such as Promote Worker or Transfer Worker, result in role provisioning and recalculation based on role provisioning rules.

When an identity’s attributes change, you may need to provision the user with different roles. Role assignments may be based on job codes, and a promotion triggers role provisioning changes. Even if the change in the identities attributes requires no role assignment change, such as with a name change, OIM synchronizes the corresponding user information in the LDAP store.

Deactivating or terminating an identity triggers revocation of some roles to end all assignments, but may provision new roles needed for activities, such as a pay stub review. If the corresponding user for the identity was provisioned with a buyer role, terminating the identity causes the user’s buyer record in Procurement to be disabled, just as the record was created when the user was first provisioned with the buyer role.

**Notifications and Audits**

Oracle Fusion Applications provides mechanisms for notifying and auditing requests or changes affecting identities and users.
Oracle Fusion Applications notifies requestors, approvers, and beneficiaries when a user account or role is provisioned. For example, when an anonymous user registers as a business-to-customer (B2C) user, the B2C user must be notified of the registration activation steps, user account, password and so on once the approver (if applicable) has approved the request and the user is registered in the system.

User ID and GUID attributes are available in Oracle Fusion Applications session information for retrieving authenticated user and identity data.

End user auditing data is stored in database WHO columns and used for the following activities.

- Setting up sign-in audit
- Using the application monitor
- Notifying of unsuccessful sign ins
- Sign-in audit reports

You can conduct real-time audits that instantiate a runtime session and impersonate the target user (with the proxy feature) to test what a user has access to under various conditions such as inside or outside firewall and authentication level.

For information on configuring audit policies and the audit store, see the Oracle Fusion Applications Administrator's Guide.

**Delegated Administration**

You can designate local administrators as delegated administrators to manage a subset of users and roles.

Delegated administrators can be internal or external persons who are provisioned with a role that authorizes them to handle provisioning events for a subset of users and roles.

For example, internal delegated administrators could be designated to manage users and roles at the division or department level. External delegated administrators could be designated to manage users and roles in an external organization such as a primary supplier contact managing secondary users within that supplier organization.

You can also define delegated administration policies based on roles. You authorize users provisioned with specific roles named in the policy to request a subset of roles for themselves if needed, such as authorizing a subset of roles for a subset of people. For example, the policy permits a manager of an Accounts Payables department to approve a check run administrator role for one of their subordinates, but prohibits the delegated administrator from provisioning a budget approver role to the subordinate.

**Credentials**

You activate or change credentials on users by managing them in Oracle Identity Management (OIM)
Applications themselves must be credentialed to access one another.

Oracle Fusion Applications distinguishes between user identities and application identities (APPID). Predefined application identities serve to authorize jobs and transactions that require higher privileges than users.

For example, a payroll manager may submit a payroll run. The payroll application may need access to the employee’s taxpayer ID to print the payslip. However, the payroll manager is not authorized to view taxpayer IDs in the user interface as they are considered personally identifiable information (PII).

Calling applications use application identities (APPID) to enable the flow of transaction control as it moves across trust boundaries. For example, a user in the Distributed Order Orchestration product may release an order for shipping. The code that runs the Pick Notes is in a different policy store than the code that releases the product for shipment. When the pick note printing program is invoked it is the Oracle Fusion Distributed Order Orchestration Application Development Framework (ADF) that is invoking the program and not the end user.

Role Provisioning and Deprovisioning: Explained

A user’s access to data and functions depends on the user’s roles: users have one or more roles that enable them to perform the tasks required by their jobs or positions. Roles must be provisioned to users; otherwise, users have no access to data or functions.

**Role Provisioning Methods**

Roles can be provisioned to users:

- Automatically
- Manually, using delegated administration:
  - Users such as line managers and human resource specialists can provision roles manually to other users.
  - Users can request roles for themselves.

For both automatic and manual role provisioning, you create a role mapping to identify when a user becomes eligible for a role.

Oracle Identity Management (OIM) can be configured to notify users when their roles change; notifications are not issued by default.

**Role Types**

Data roles, abstract roles, and job roles can be provisioned to users. Roles available for provisioning include predefined roles, HCM data roles, and roles created using OIM.

**Automatic Role Provisioning**

A role is provisioned to a user automatically when at least one of the user’s assignments satisfies the conditions specified in the relevant role-mapping
definition. The provisioning occurs when the assignment is either created
or updated. For example, when a person is promoted to a management
position, the line manager role is provisioned automatically to the person if an
appropriate role mapping exists. Any change to a person’s assignment causes
the person’s automatically provisioned roles to be reviewed and updated as
necessary.

**Role Deprovisioning**

Automatically provisioned roles are deprovisioned automatically as soon as
a user no longer satisfies the role-mapping conditions. For example, a line
manager role that is provisioned to a user automatically is deprovisioned
automatically when the user ceases to be a line manager.

Automatically provisioned roles can be deprovisioned manually at any time.

Manually provisioned roles are deprovisioned automatically only when all of the
user’s work relationships are terminated; in all other circumstances, users retain
manually provisioned roles until they are deprovisioned manually.

**Changes to Assignment Managers**

When a person’s line manager is changed, the roles of both new and previous
line managers are updated as necessary. For example, if the person’s new line
manager now satisfies the conditions in the role mapping for the line manager
role, and the role is one that is eligible for autoprovisioning, then that role is
provisioned automatically to the new line manager. Similarly, if the previous line
manager no longer satisfies the conditions for the line manager role, then that
role is deprovisioned automatically.

**Roles at Termination**

When a work relationship is terminated, all automatically provisioned roles for
which the user does not qualify in other work relationships are deprovisioned
automatically. Manually provisioned roles are deprovisioned automatically
only if the user has no other work relationships; otherwise, the user retains all
manually provisioned roles until they are deprovisioned manually.

Automatic deprovisioning can occur either as soon as the termination is
submitted or approved or on the day after the termination date. The user who is
terminating the work relationship selects the appropriate deprovisioning date.

Role mappings can provision roles to users automatically at termination. For
example, the locally defined roles Retiree and Beneficiary could be provisioned
to users at termination based on assignment status and person type values.

If a termination is later reversed, roles that were deprovisioned automatically
at termination are reinstated and post-termination roles are deprovisioned
automatically.

**Date-Effective Changes to Assignments**

Automatic role provisioning and deprovisioning are based on current data. For a
future-dated transaction, such as a future promotion, role changes are identified
and role provisioning occurs on the day the changes take effect, not when the
change is entered. The process Send Pending LDAP Requests identifies future-
dated transactions and manages role provisioning and deprovisioning at the
appropriate time. Note that such role-provisioning changes are effective as of the
system date; therefore, a delay of up to 24 hours may occur before users in other
time zones acquire the access for which they now qualify.

**Role Mappings: Explained**

User access to data and functions is determined by abstract, job, and data roles,
which are provisioned to users either automatically or manually. To enable a
role to be provisioned to users, you define a relationship, known as a mapping,
between the role and a set of conditions, typically assignment attributes such
as department, job, and system person type. In a role mapping, you can select
any role stored in the Lightweight Directory Access Protocol (LDAP) directory,
including Oracle Fusion Applications predefined roles, roles created in Oracle
Identity Management (OIM), and HCM data roles.

The role mapping can support:

- Automatic provisioning of roles to users
- Manual provisioning of roles to users
- Role requests from users
- Immediate provisioning of roles

**Automatic Provisioning of Roles to Users**

A role is provisioned to a user automatically if:

- At least one of the user's assignments satisfies all conditions associated
  with the role in the role mapping.
- You select the **Autoprop** option for the role in the role mapping.

For example, for the HCM data role Sales Manager Finance Department, you
could select the **Autoprop** option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>Finance Department</td>
</tr>
<tr>
<td>Job</td>
<td>Sales Manager</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

The HCM data role Sales Manager Finance Department is provisioned
automatically to users with at least one assignment that satisfies all of these
conditions.

Automatic role provisioning occurs as soon as the user is confirmed to satisfy
the role-mapping conditions, which can be when the user's assignment is either
created or updated. The provisioning process also removes automatically
provisioned roles from users who no longer satisfy the role-mapping conditions.

**Note**

The automatic provisioning of roles to users is effectively a request to OIM to
provide the role. OIM may reject the request if it violates segregation-of-duties
rules or fails a custom OIM approval process.
Manual Provisioning of Roles to Users

Users such as human resource (HR) specialists and line managers can provision roles manually to other users; you create a role mapping to identify roles that can be provisioned in this way.

Users can provision a role to other users if:

- At least one of the assignments of the user who is provisioning the role (for example, the line manager) satisfies all conditions associated with the role mapping.
- You select the Requestable option for the role in the role mapping.

For example, for the HCM data role Quality Assurance Team Leader, you could select the Requestable option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies both of these conditions can provision the role Quality Assurance Team Leader manually to other users, who are typically direct and indirect reports.

If the user’s assignment subsequently changes, there is no automatic effect on roles provisioned by this user to others; they retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

Role Requests from Users

Users can request roles when reviewing their own account information; you create a role mapping to identify roles that users can request for themselves.

Users can request a role if:

- At least one of their own assignments satisfies all conditions associated with the role mapping.
- You select the Self-requestable option for the role in the role mapping.

For example, for the Expenses Reporting role you could select the Self-requestable option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>ABC Department</td>
</tr>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies all of these conditions can request the role. The user acquires the role either immediately or, if approval is required, once the request is approved. Self-requested roles are classified as manually provisioned.
If the user’s assignment subsequently changes, there is no automatic effect on self-requested roles. Users retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

**Immediate Provisioning of Roles**

When you create a role mapping, you can apply autoprovisioning from the role mapping itself.

In this case, all assignments and role mappings in the enterprise are reviewed. Roles are:

- Provisioned immediately to all users who do not currently have roles for which they are eligible
- Deprovisioned immediately from users who are no longer eligible for roles that they currently have

Immediate autoprovisioning from the role mapping enables bulk automatic provisioning of roles to a group of users who are identified by the role-mapping conditions. For example, if you create a new department after a merger, you can provision relevant roles to all users in the new department by applying autoprovisioning immediately.

To provision roles immediately to a single user, the user’s line manager or an HR specialist can autoprovision roles from that user’s account.

**Role-Mapping Names**

The names of role mappings must be unique in the enterprise. You are recommended to devise a naming scheme that reveals the scope of each role mapping. For example:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autopropvisioned Roles Sales Department</td>
<td>Mapping includes all roles provisioned automatically to anyone in the sales department</td>
</tr>
<tr>
<td>Benefits Specialist Autopropvisioned</td>
<td>Mapping defines the conditions for autoprovisioning the Benefits Specialist role</td>
</tr>
<tr>
<td>Line Manager Requestable Roles</td>
<td>Mapping includes all roles that a line manager can provision manually to direct and indirect reports</td>
</tr>
</tbody>
</table>

**Role Mappings: Examples**

Roles must be provisioned to users explicitly, either automatically or manually; no role is provisioned to a user by default. This topic provides some examples of typical role mappings to support automatic and manual role provisioning.

**Creating a Role Mapping for Employees**

You want all employees in your enterprise to have the Employee role automatically when they are hired. In addition, employees must be able to request the Expenses Reporting role when they need to claim expenses. Few employees will need this role, so you decide not to provision it automatically to all employees.
You create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

- Employee role, and select the **Autoprovision** option
- Expenses Reporting role, and select the **Self-requestable** option

You could create a similar role mapping for contingent workers called All Contingent Workers, where you would set the system person type to contingent worker.

**Note**

If the Employee and Contingent Worker roles are provisioned automatically, pending workers acquire them when their periods of employment or placements start. If they need roles before then, you create a separate role mapping for the pending worker system person type.

**Creating a Role Mapping for Line Managers**

Any type of worker can be a line manager in the sales business unit. You create a role mapping called Line Manager Sales BU and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Unit</td>
<td>Sales</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
</tbody>
</table>

You include the Line Manager role and select the **Autoprovision** option. This role mapping ensures that the Line Manager role is provisioned automatically to any worker with at least one assignment that matches the role-mapping conditions.

In the same role mapping, you could include roles that line managers in this business unit can provision manually to other users by selecting the roles and marking them as requestable. Similarly, if line managers can request roles for themselves, you could include those in the same role mapping and mark them as self-requestable.

**Creating a Role Mapping for Retirees**

Retirees in your enterprise need a limited amount of system access to manage their retirement accounts. You create a role mapping called All Retirees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Retiree</td>
</tr>
</tbody>
</table>
You include the locally defined role Retiree in the role mapping and select the **Autoprovison** option. When at least one of a worker's assignments satisfies the role-mapping conditions, the Retiree role is provisioned to that worker automatically.

### Creating a Role Mapping for Sales Managers

Grade 6 sales managers in the sales department need the Sales Manager role. In addition, sales managers need to be able to provision the Sales Associate role to other workers. You create a role mapping called Sales Managers Sales Department and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>Sales</td>
</tr>
<tr>
<td>Job</td>
<td>Sales manager</td>
</tr>
<tr>
<td>Grade</td>
<td>6</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping, you include the:

- Sales Manager role, and select the **Autoprovison** option
- Sales Associate role, and select the **Requestable** option

### Defining Security After Enterprise Setup: Points to Consider

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Import Worker Users
- Import Partner Users
- Manage Job Roles
- Manage Duties
- Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human Capital Management (HCM), Supplier Portal, and Customer Relationship Management (CRM) Self Service, respectively, and associate them with users.
Before Importing Users

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Load Batch Data task. If user and role provisioning rules have been defined, the Load Batch Data process automatically creates user and role provisioning requests as the workers are created.

Once the enterprise is set up, performing the Load Batch Data task populates the enterprise with HR workers in records linked by global user ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Load Batch Data task results in new user accounts being created. Worker email addresses as an alternate input for the Load Batch Data task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Load Batch Data task in HCM.

Note

The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in CRM.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

Import Users

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks.

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Load Batch Data task to create users from imported HR workers.
Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager.

In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

**Manage Job Roles**

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

**Note**

Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, CRM or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

**Manage Duties**

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit
duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering.

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.

In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

**Tip**

As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

**Note**

Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

**Manage Application Access Controls**

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

**Note**

In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is...
Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

Creating Users: Worked Example

You can create users by entering basic person and employment data. A user account is created automatically for a person when you create the user record. You can assign the users Oracle Fusion Human Capital Management (HCM) and non-HCM data roles, each providing access to specific functions and data. This example demonstrates how to create a user and assign roles to the user.

Note

This user management functionality is available for HCM Foundation and Oracle Fusion Workforce Directory Management (WDM) users only.

Decisions to Consider

<table>
<thead>
<tr>
<th>For whom are you creating the user record?</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gail Williams</td>
<td>Gail Williams</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What is the user account name?</th>
<th>Same as the e-mail ID, <a href="mailto:gail.williams@vision.com">gail.williams@vision.com</a></th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Where is Gail employed?</th>
<th>Gail is an employee of Vision Corporation, and works in the Human Resources (HR) department in the Canada office.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>What roles must be provisioned to Gail?</th>
<th>Autoprovision the employee role. Gail is responsible for processing workers’ expense claims so provision the role Expense Claims Administrator manually to Gail.</th>
</tr>
</thead>
</table>

Prerequisites

1. Create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

- Employee role, and select the Autoprovision option

- Expense Claims Administrator role, and select the Self-requestable option

Creating a User

1. On the Manage Users page, in the Search Results region, click the Create icon to open the Create User page.
2. Complete the fields, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>Williams</td>
</tr>
<tr>
<td>First Name</td>
<td>Gail</td>
</tr>
<tr>
<td>E-Mail</td>
<td><a href="mailto:gail.williams@vision.com">gail.williams@vision.com</a></td>
</tr>
<tr>
<td>Hire Date</td>
<td>4/12/11</td>
</tr>
</tbody>
</table>

3. In the User Details region, leave the User Name field blank. The user name defaults to the user’s e-mail ID.

4. In the Employment Information region, select the person type **Employee** and complete the fields as shown in the table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Legal Employer</td>
<td>Vision Corporation</td>
</tr>
<tr>
<td>Business Unit</td>
<td>Vision Canada</td>
</tr>
<tr>
<td>Department</td>
<td>Human Resources</td>
</tr>
</tbody>
</table>

**Assigning Roles to the User**

1. Click **Autoprovision Roles** to provision the employee role to the user.
2. Click **Add Role**.
3. Search for and select the **Expense Claims Administrator** role.
4. Click **Save and Close**. The user account is created and the roles are assigned to the user immediately.

**Define Automated Governance, Risk, and Performance Controls**

**Segregation of Duties: Explained**

Segregation of duties (SOD) separates activities such as approving, recording, processing, and reconciling results so an enterprise can more easily prevent or detect unintentional errors and willful fraud. SOD policies, called access control policies in Application Access Controls Governor (AACG), exert both preventive and detective effects.

SOD policies constrain duties across roles so that unethical, illegal, or damaging activities are less likely. SOD policies express constraints among roles. Duty role definitions respect segregation of duties policies.

**Application Access Controls Governor**

You manage, remediate, and enforce access controls to ensure effective SOD using the Application Access Controls Governor (AACG) product in the Oracle Governance, Risk and Compliance Controls (GRCC) suite.
AACG applies the SOD policies of the Oracle Fusion Applications security reference implementation using the AACG Oracle Fusion Adapter.

AACG is integrated with Oracle Identity Management (OIM) in Oracle Fusion Applications to prevent SOD control violations before they occur by ensuring SOD compliant user access provisioning. SOD constraints respect provisioning workflows. For example, when provisioning a Payables role to a user, the SOD policy that ensures no user is entitled to create both an invoice and a payment prevents the conflicting roles from being provisioned. AACG validates the request to provision a user with roles against SOD policies and provides a remediating response such as approval or rejections if a violation is raised.

Use AACG to for the following.

- Define SOD controls at any level of access such as in the definition of an entitlement or role.
- Simulate what-if SOD scenarios to understand the effect of proposed SOD control changes.
- Use the library of built-in SOD controls provided as a security guideline.

Managing Segregation of Duties

SOD policies express incompatible entitlement or incompatible access points into an application. In GRCC, an access point is the lowest level access for a particular application. In GRCC, entitlement is a grouping of access points. As a security guideline, group the lowest level access points or define the SOD policy at the access level causing the least amount of change. Business activities are enabled at access points. In Oracle Fusion Applications, the hierarchy of access points in descending levels is users, roles, and entitlement.

Note

AACG entitlements are logical groupings of security objects that represent Oracle Fusion Application access points such as roles or entitlement.

Note

In AACG, segregation of duties policies are called access controls.

Oracle Fusion Applications does not predefine business logic for dealing with SOD conflicts. Oracle Fusion Applications does define a set of states where role requests are suspended pending resolution of SOD violations the role request introduces. In most cases, Oracle Fusion Applications invokes OIM to handle role requests. Enterprises define SOD resolution rules when defining SOD policy.

Remediating Segregation of Duties Policy Violations

The risk tolerance of your enterprise determines what duties must be segregated and how to address violations.

AACG assists in remediation of violations with a guided simulation that identifies corrective action. You determine the exact effects of role and
Defining Segregation of Duties Policies: Points To Consider

Segregation of duties (SOD) policies express incompatibilities enforced to control access in defined contexts.

In Oracle Fusion Applications, SOD policies protect against the following incompatibilities.

- Privilege X is incompatible with privilege Y
- Role A is incompatible with role B
- Any privileges in role A are incompatible with any privileges in role B.
- Privilege X is incompatible with any privileges in role B.

The following examples of SOD policies illustrate incompatible entitlement.

- No user should have access to Bank Account Management and Supplier Payments duties.
- No user should have access to Update Supplier Bank Account and Approve Supplier Invoice entitlement.

Data Contexts

You can extend SOD policies to control access to specific data contexts.

For example, no single individual must be able to source a supplier in a business unit and approve a supplier invoice in the same business unit.

Exclusion and Inclusion Conditions

SOD policies may include exclusion conditions to narrow the SOD scope and reduce false positive violations, or inclusion conditions to broaden the scope.

Conditions apply to access points globally, to policies, or to access paths defined by policies. Access path conditions can exclude a user from a role, an Oracle Fusion Applications entitlement from a role, or a permission from an Oracle Fusion Applications entitlement.

The following global exclusion conditions are predefine in Oracle Fusion Applications and available when creating SOD policies.

- User Status
- User Name

For information on managing segregation of duties, see the Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User's Guide.
• Enterprise Role
• Action
• Business Unit
• Within Same Business Unit

**Enforcement**

Oracle Fusion Applications enforces SOD policies under the following circumstances.

• When granting entitlement to a role
• When provisioning a role to a user

For information on managing segregation of duties, see Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User's Guide.

---

**Note**

SOD policies are not enforced at the time of role definition.

---

A single SOD policy can include entitlement from multiple instances of a single enterprise resource planning environment. For example, one SOD policy is enforced in implementation, test, and production instances of Oracle Fusion Applications.

---

**Managing Segregation of Duties Risks and Violations: Critical Choices**

You assess and balance the cost of duty segregation against reduction of risk based on the requirements of your enterprise.

The types of people who resolve SOD conflicts include the following.

• Administrator of an external program such as the Procurement Administrator for the supplier portal or the Partner Manager for the PRM Program
• Senior executive spanning multiple organizations in an enterprise with opposing interests
• Risk management professional implementing a Oracle Governance, Risk and Compliance Controls (GRCC) initiative
  • Predefines a set of conditions and informs access provisioning staff to approve requests and prove the exception based on certain conditions
  • Allows defining rules to route SOD violations for approval
You view and respond to risks and violations in the Application Access Controls Governor (AACG).

You may wish to override an SOD violation. For example, the Accounts Payable Supervisor includes incompatible duties to create both invoices and payments when you provision this job role to a user, you may waive the violation in the AACG. You may waive the violation for the currently provisioned user, for the SOD policy that raised the violation, or for the SOD policy within a particular data set, such as a business unit.

The risk tolerance of your enterprise guides how you respond to conflicts. For example, a user may be provisioned with both the role of Order Manager and Shipping Agent. The Order Manger role entitles the user to enter orders, which could result in exploitation when filling shipping quotas. You can remove the entitlement to enter orders that the Order Manger job role inherits from the Orchestration Order Scheduling Duty role. Or you could segregate the shipping and order entry duties by defining an SOD policy that allows a user to have either job role but not both.

False Positives

False positives can be SOD policy violations that aren’t actually violations or violations that are within your risk tolerance and therefore do not require corrective action.

You can reduce false positives by the following methods.

- Define exclusion conditions that can be applied to individual or groups of policies.
- Define logically complex SOD policies that enforce more exacting specifications.
- Determine whether conflicts should be prevented, monitored, or made subject to approval during provisioning.

Path Level Detection

Conflict analysis detects a user’s multiple paths to one or more conflicting access points.

For example, a user may be able to reach a single access point through one or more roles, or by one entitlement leading to another through submenus to a function that represents a risk. The resulting conflict path shows if the conflict is generated by inappropriate role provisioning or configuration of applications. The audit shows the paths from any number of users to any number of access points involved in conflicts, which lets you visualize the root cause and remediate effectively.

AACG assigns one or more users to review all paths involved in a given conflict so that the entire conflict can be addressed in a coherent way.

Waiving or Accepting Violations

AACG lets you accept or waive a violation. Your reasons may include that you accept the risk or will define compensating controls.
A waiver may apply to the current user, constraint, or constraint within a dimension such as the business unit.

**Resolving Conflicts**

The risk tolerance of the enterprise determines whether a segregation of duties conflict must be removed from the security reference implementation.

The following approaches resolve conflicts.

- Change the segregation of duties policy.
- Ensure a job role does not contain incompatible duties.
- Define data security policies that restrict authorized access by incompatible duties.

Changing a segregation of duties policy may not be possible in most cases. For example, a policy that segregates creation of payables invoice from making payables payments should be preserved, even if the Accounts Payables Manager job role includes a duty role for each activity. To prevent an accounts payables manager from being authorized to perform both duties, or from being authorized to make payables payments to self and direct reports, the Accounts Payables Manager job role must be changed. The security implementation can be changed to include two job roles that segregate the incompatible duties. Added data security policy grants can restrict the access to at risk data.

For information on managing segregation of duties, see the Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User’s Guide.

**Role Provisioning and Segregation of Duties: How They Work Together**

Segregation of duties (SOD) checks occur when roles are assigned to users. The checks are based on Oracle Application Access Controls Governor (AACG) policies. The Oracle Identity Management (OIM) integration includes predefined routing rules for remediation in the Manage IT Security business process.

External users such as suppliers or partners need to be provisioned with roles to facilitate access to parent company interfaces and data. The process by which such provisioning requests are approved in Oracle Fusion Applications helps explain the request flows and possible outcomes.

**Note**

In Oracle Identity Management (OIM), external users means users who are not specific to applications, such as enterprise roles or the absence of entitlement to access an application.

The figure shows the role provisioning request flow. OIM uses AACG to check segregation of duties violations.
Tables

A supplier or partner requests admission to a program using an implementation of the Supplier Portal Submission. The submission is captured in one or both of the following tables in advance of approving or rejecting the supplier or partner.

- Oracle Fusion Trading Community Model
- Interface Staging

Oracle Fusion Applications collects the employee names for the supplier or partner company at the time the company submits its request to join the program so that all employees accessing Oracle Fusion Applications on behalf of the supplier or partner are provisioned.

AACG in the Oracle Governance, Risk and Compliance Controls (GRCC) suite is certified to synchronize with the policy and identity stores for all pillars or partitions of Oracle Fusion Applications and integrated with the Oracle Fusion Applications security approach to roll up entitlements (by means of duty roles) to the roles that are provisioned to internal users. SOD policies can be defined and enforced at any level of authorization. For external users, SOD policies use attribute information stored in the Trading Community Model tables.
OIM and the SPML Client

Enterprise business logic may qualify the requester and initiate a role provisioning request by invoking the Services Provisioning Markup Language (SPML) client module, as may occur during onboarding of internal users with Human Capital Management (HCM), in which case the SPML client submits an asynchronous SPML call to OIM. Or OIM handles the role request by presenting roles for selection based on associated policies.

OIM recognizes the role provisioning request and initiates a call to AACG.

OIM apprises the SPML client of the current state of the role provisioning request as SOD_CHECK_IN_PROGRESS.

OIM stores the SOD check result as part of OIM audit data.

OIM apprises SPML client of the current state of the SPML request. The provisioning is either still in progress with segregation of duties being checked, or conflicts were found. If conflicts exist, AACG rejects the request and notifies the application.

<table>
<thead>
<tr>
<th>Status</th>
<th>Conflicts</th>
<th>Current State</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOD_CHECK_IN_PROGRESS</td>
<td>Unknown</td>
<td>Request sent to AACG and waiting for response</td>
</tr>
<tr>
<td>SOD_REMEDIATION_IN_PROGRESS</td>
<td>Conflict found</td>
<td>AACG detected violations and remediation is in progress</td>
</tr>
<tr>
<td>SOD_CHECK_APPROVED</td>
<td>No conflict found</td>
<td>No SOD violations found</td>
</tr>
<tr>
<td>SOD_CHECK_REJECTED</td>
<td>Conflict found</td>
<td>AACG detected violations that cannot be remediated</td>
</tr>
<tr>
<td>SOD_REMEDIATION_APPROVED</td>
<td>Conflict found</td>
<td>AACG detected violations that are approved</td>
</tr>
<tr>
<td>SOD_REMEDIATION_REJECTED</td>
<td>Conflict found</td>
<td>AACG detected violations that are rejected by approver</td>
</tr>
</tbody>
</table>

In the absence of an SOD exception, OIM provisions all relevant users.

Note

When a partner user is provisioned, all employees of the partner enterprise are provisioned. SOD checks occur when an external user requests to join a program, because SOD policies operate across Oracle Fusion Applications, not at the individual level. Supplier or partner company user requests are not approved if there is an SOD conflict against the supplier company.

OIM provides AACG with the details of SOD exception approval workflow. AACG audits the outcome for use in future detective controls and audit processes.

Oracle Application Access Controls Governor

AACG may respond with the following.

- Roles may be provisioned to the external user or its employees because no SOD conflict is found
• SOD conflict is found and request is denied because the relevant SOD
policy is to be strictly enforced and no exception approval should be
allowed

• SOD conflict is found and the exception to the policy is allowed, so the
request goes through additional processing, such as an approval process.

Supplier or Partner Relationship Management responds to an SOD exception by
updating Trading Community Model tables with the current state. An enterprise
may elect to implement a landing pad that offers external users a means of
addressing the SOD problem by providing more information or withdrawing the
request.

SOD violation checking occurs during role implementation and provisioning,
and can be turned on or off if AACG is provisioned and enabled as part of the
Oracle Fusion Applications deployment.

**Segregation of Duties Exception Resolution or Approval Workflow**

Depending upon status, OIM kicks off an auditable SOD exception resolution
workflow. Resolution can be conditional based on approval or requirements such
as contracts being met.

If one of the paths for exception resolution is to get an approval, then the SOD
exception resolution drives the approval using AMX. Standard AMX rules,
not business rules, resolve the approval for the SOD exception, including the
following.

- Organizational hierarchies
- Multiple mandatory and optional approvers
- Rerouting and approval delegation

The approver resolution uses AMX Rules Designer to access various user
attributes and organizational hierarchies managed in Oracle Fusion Applications
repositories. This information is typically not available in OIM or the LDAP
identity store repository. Enterprises can define additional approval rules using
AMX Thin Client.

The SOD Exception Approver gets a notification through supported channels
that a new request is awaiting approval. The approver signs in to the global SOA
federated worklist application that aggregates all pending worklist items for
the user from all Oracle Fusion applications and logical partitions or pillars of
applications. The SOD exception approval tasks show up in the same list.

The SOD exception approval task shows the details of the SPML request and
SOD Provisioning results in a page rendered by OIM. The approver may take
one of the following actions.

- Approve the request as it is
- Reject the request

If the approver approves the request, OIM sends an
SOD_REMEDIATION_APPROVED status to the SPML client.

If the approver rejects the request, OIM sends an
SOD_REMEDIATION_REJECTED status to the SPML client. The provisioning
request is considered completed with a failure outcome and the external users
is notified. Oracle Fusion Applications updates the Trading Community Model tables with the rejected status

**Remediation Task Assignments**

The SOD remediation tasks are assigned based on the role being requested.

1. If the role requested is Chief Financial Officer, the SOD remediation task is assigned to the IT Security Manager role.

2. If the SOD violation results from a policy where the SOD control tag is the Information Technology Management business process and the control priority is 1, the SOD remediation task is assigned to the Application Administrator role.

3. In all other scenarios, the SOD remediation task is assigned to the Controller role.

For more information about configuring audit policies, see the Oracle Fusion Applications Administrator’s Guide.

For information on managing segregation of duties, see the Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User’s Guide.

---

**Define Approval Management for Customer Relationship Management**

**Approval Management: Highlights**

Use approval management to determine the policies that apply to approval workflows for particular business objects such as expense reports. For example, you can specify levels of approval for expense reports over a particular amount, to reflect your own corporate policies. You also determine the groups of users who act on these workflow tasks, for example, the chain of approvers for expense reports.

Approval management is fully described in the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management. Though the concepts described there apply also to Oracle Fusion Applications, the only setup relevant to Oracle Fusion Applications involves approval groups and task configuration. Customization of approval workflows themselves is described in the Oracle Fusion Applications Extensibility Guide.

**Overview**


See: Introduction to Approval Management

See: Understanding Approval Management Concepts
Approval Groups and Task Configuration

- An approval group consists of a name and a predefined set of users configured to act on a task in a certain pattern. Refer to the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management.

  See: Administering Approval Groups


  See: Using Task Configuration

Customization

- You can optionally customize predefined approval workflows, for example add post-approval activities or additional stages. Refer to the Oracle Fusion Applications Extensibility Guide.

  See: Customizing and Extending SOA Components

Define Help Configuration

Define Help Configuration: Overview

The Define Help Configuration task list contains tasks that let you set up and maintain Oracle Fusion Applications Help for all users, if you have a local installation of help. Use the Set Help Options task to determine if certain aspects of Oracle Fusion Applications Help are available to users and to control how aspects of the help site work. Use the Assign Help Text Administration Duty and Manage Help Security Groups tasks to set up customization of help content.

After performing the help configuration tasks, you can review the predefined help and consider whether to add or customize any content. You can also customize help that is embedded in the application, for example hints and help windows, using other tools such as Oracle JDeveloper and Oracle Composer.

Use the Setup and Maintenance work area to access the tasks in the Define Help Configuration task list.

Help Feature Choices and Help Options: Points to Consider

Help feature choices on the Configure Offerings page in the Setup and Maintenance work area control the look and behavior of Oracle Fusion Applications Help, and also determine which help options are available. Help options are setup options on the Set Help Options page, and they pertain only to local installations of help.

Local Installation of Help

Select the Local Installation of Help feature choice if you want to customize help, or if you do not want users to go outside your firewall to access the public help
site on the Internet. If you have Oracle User Productivity Kit (UPK), then you can also incorporate custom UPK content into a local help system. Announcements and discussions would be available only to users of the local installation of help. Make sure that maintaining a server for local help is not an issue.

Do not install help locally if you do not need to customize help, or do not want to or cannot maintain and monitor resources for help on your own. If you also have a fast connection to the Web, then consider using the public help site instead. Users of the public site would see announcements from Oracle, and all discussions are public and moderated by Oracle.

If you select the Local Installation of Help feature choice, then the Define Help Configuration task list appears in your implementation project, and you can select two additional features (Access to Internet-Based Help Features and Help Customization) to control the fields available on the Set Help Options page.

**Access to Internet-Based Help Features**

Select this feature choice to provide users of a local help installation access to features that involve integration with the public help site or navigation to other sites on the Web. Consider if users would benefit from, for example, seeing help file ratings from the public help site. Do not select this feature if you do not want users going outside your firewall for any reason.

**Important**

Some help includes links to the Oracle Fusion Applications Technology Library. If you select this feature, then these links open the library on the Oracle Technology Network Web site. If you do not select this feature, then you must apply the Technology Library patch from My Oracle Support so that users can access the library locally.

If you select this feature choice, then these help options are available:

- Share ratings and comments with Internet-based help site
- Display average ratings from Internet-based help site
- All options in the Web Sites Available from Help Site section

**Help Customization**

Select the Help Customization feature choice if you intend to customize predefined help or add your own files to your local installation of help. For example, you can add internal policies or procedures as help, and Oracle User Productivity Kit content, if any. Not selecting this feature choice only hides the relevant setup options; help customization is not disabled for users with customization access. To disable help customization, remove the Oracle Fusion Help Text Administration duty from the job roles where it is predefined.

If you select this feature choice, then the Custom Help Security feature choice is available, as well as all these sections on the Set Help Options page:

- Custom Help
- User Productivity Kit
Custom Help Security

Select this feature choice if you want certain help files to be available only to a restricted set of users. You can define the user groups allowed to view corresponding help files. Do not select this feature choice if you do not have this requirement, because the feature can have an impact on performance.

If you select the Custom Help Security feature choice, then the Manage Help Security Groups task is available in the Define Help Configuration task list in your implementation project. There are no help options associated with this feature choice.

Administering Collaboration Features and Announcements in Help: Points to Consider

Announcements and collaboration features (discussions, ratings and comments) allow users to share information regarding help and the subjects that particular help files cover. These collaboration features are also used elsewhere in Oracle Fusion Applications. If you have a local installation of Oracle Fusion Applications Help, then you can use the Set Help Options page in the Setup and Maintenance work area to enable the announcements and discussions features and to set options about ratings. When administering these features, consider the purpose of each feature and points that are specific to Oracle Fusion Applications Help.

Announcements

Use announcements to broadcast information to all users of your locally installed help site. You can provide information about help, for example new custom help that was recently added, or about anything that users should take note of, for example a change in company policy. Announcements can appear on any of the tabs on the home page of Oracle Fusion Applications Help. You can target specific user groups by posting announcements to specific tabs, for example, posting information related to implementation to the Functional Setup tab.

Only administrators for discussions can post announcements to the help site. For more information on granting administrator roles for discussions, see the Oracle Fusion Middleware Administrator’s Guide for Oracle WebCenter. The public help site on the Internet contains announcements from Oracle.

Discussions

Users can use discussions to post questions or comments about subjects covered in specific help files. For example, after reading help on expense reports, users might have questions or comments about company policies or processes for expenses. Other users who later access this help file would benefit from the information in the discussion.

If you have a local installation of help, then you can set a help option to enable discussions, and assign specific users to moderate discussions. Each help file would contain a Discuss link that all users across your site can use to read
discussions about that file. They can also start a discussion topic or post to existing topics. These discussions are visible only to users of your locally installed help site.

**Important**

Do not enable discussions until servers for discussions are up and running.

The public help site always has discussions enabled and available to any user. These discussions are visible to the general public and moderated by Oracle.

**Ratings and Comments**

Users can rate any help file on a five star system and provide feedback about the content. This information is helpful to other users in deciding which help file to open. Help files with a higher average rating are listed first in help windows, and in the help listings you see as you browse using the help navigators.

In local installations of help, you can set help options to share ratings with the public help site or display ratings from the public site. Otherwise, the scope of ratings and reviews is limited to your local installation.

**Creating Help Security Groups: Worked Example**

This example demonstrates how to create a help security group to define a set of job roles that have access to help. The help security group can then be assigned to particular help files so that only users with any of the defined roles have access to the help.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What type of users do you need to limit help access to?</td>
<td>Human resources (HR) specialists</td>
</tr>
<tr>
<td>Is there a specific time period for which this access is needed?</td>
<td>No, the help files should always be viewed only by the HR specialists</td>
</tr>
<tr>
<td>Where do you want this group to appear in the list of values for help security groups?</td>
<td>First</td>
</tr>
</tbody>
</table>

Define a help security group and assign a duty role to the group.

1. From the Setup and Maintenance work area, find the Manage Help Security Groups task and click **Go to Task**.
3. Complete the fields, as shown in this table. Leave the start and end dates blank.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Help Security Group</td>
<td>HR</td>
</tr>
<tr>
<td>Meaning</td>
<td>HR Only</td>
</tr>
</tbody>
</table>
4. Click **Save**.

5. With your new help security group selected, go to the Associated Roles section and add a new row.

6. Select **PER_HUMANRESOURCE_SPECIALIST** as the role name.

7. Click **Save and Close**.

You have created a new lookup code for the Help Security Groups lookup type, which is a standard lookup. The lookup code has the name, meaning, and description that you defined for the help security group.

You have also created a data security policy for the help database resource, specifying that the Human Resource Specialist role can view help that is defined with the HR security group. If you go to the Manage Database Resources and Policies page and find the database resource, or object, ATK_KR_TOPICS, then you can see the policy for the Human Resource Specialist role, with the condition that the column name, SECURITY_CODE, is equal to the value HR.

### Help File Customization: Overview

If you have a local installation of Oracle Fusion Applications Help and the appropriate job roles, then you can customize the help files in the help site. Use the Manage Custom Help page to maintain both predefined and custom help files. You can create, duplicate, edit, and delete custom files, or set their status to Active or Inactive. For predefined files, you can only duplicate them or set their status. For each help file, predefined or custom, use help locations to determine where the help file appears in the application and in the help site. You have various options in how you add custom help, for example by uploading a file or specifying a URL.

Many help files can be accessed from help windows in the application. If you want to customize help in the context of a help window, for example create a custom help file and add a link to it from a specific help window, then start by opening that help window. When you click the *Manage Custom Help* link, you go to the Manage Custom Help page, and the help location fields are automatically populated with values that correspond to the help window. This way you can easily select existing files to add to the same help location, and when you create a new file, the same help location appears by default.

You can also open the Manage Custom Help page directly from the home page of Oracle Fusion Applications Help. To edit a specific file, you can either find it in the Manage Custom Help page, or open the file itself and click the *Edit* link.

---

**Note**

When you search in the Manage Custom Help page, make sure that the **Custom Help Only** check box is not selected if you are looking for predefined help.
Help Locations: Explained

Help locations determine where users can find help files, custom or not, from either the application or the help site.

Help locations include:

- Page or section values
- Help hierarchies
- Primary locations

Page or Section Values

The value in the **Page or Section** field on the help customization pages represents where users can click a help icon to open a help window that contains a link to the help file. In most cases, this value represents a page or region header in the application. Help windows are also available on specific tabs or windows, and in the Setup and Maintenance work area for specific task lists or tasks. You can associate a help file with multiple page or section values, or with none at all.

The page or section value reflects the logical navigation to the help window. For example, **Edit Opportunity page, Revenue tab, Recommendations window** does not mean that the help file is available in three different places. The help icon is in the Recommendations window, which is accessed from the Revenue tab on the Edit Opportunity page.

If the value suggests multiple locations, for example **Create and Edit Opportunity pages**, then the help file is available from the page header of both the Create Opportunity and Edit Opportunity pages. If the page or section value is, for example, a dashboard region that appears in multiple dashboards, then the value does not specify the page name but just the region. The help file is available from that region in multiple dashboards.

Help Hierarchies

Help files are associated with help hierarchies, which are used to categorize help files and aid users in finding help. Each help file can have multiple hierarchies, with at least one of type Business Processes. The business process hierarchy is based on the Business Process Management model. Every page or section value is predefined with a specific business process hierarchy. If you select a page or section without entering a business process hierarchy, the predefined hierarchy appears by default.

The Search by Business Process navigator in the help site is based on the business process hierarchy. For example, if you assign two business process hierarchies to a help file, users can find the file in both locations in the navigator. When the user clicks **More Help** from a help window, all help files assigned to the same business process hierarchy as the page or section value are returned as search results.

Similarly, the Search by Product navigator is based on the Product hierarchy type, in which level 1 is the product family, level 2 is the product, and level 3 is the business activity owned by that product.
The Search by Functional Setup navigator is based on the Functional Setup hierarchy type. The level 1 nodes for this hierarchy are:

- Functional Setup Manager, which includes help about using the Setup and Maintenance work area.
- Guides, which contains level 2 nodes that correspond to business areas and setup offerings. All the user reference and functional setup PDF guides are included.
- Offerings, which contains level 2 nodes for each setup offering, and lower levels for the main task lists in the offerings. Help for the task lists and tasks are included.

The Search by Common Tasks navigator is based on the Welcome hierarchy type. The level 1 nodes represent categories of functional areas common to all users.

Primary Locations

The primary location of a help file designates the hierarchy that is displayed for the help file in search results and within the help content as breadcrumbs. You cannot change the primary location of a help file that came with your help installation. Primary locations of predefined help are based on the business process hierarchy, while custom help files can have primary locations based on hierarchies of any type.

Editing Predefined Help: Points to Consider

When you open any predefined help file that came with your local installation of Oracle Fusion Applications Help, you can see an edit option if you have roles allowing edit access. To edit predefined help, consider:

- What happens to the original help file
- Where predefined help appears

What Happens to the Original Files

When you edit predefined help, you are actually creating a new custom help file based on the original file, with the same help locations. The customized version replaces the original, which becomes inactive and hidden from users. You can display both versions by reactivating the original in the Manage Custom Help page.

Note

In the Search Results: Existing Help region on the Manage Custom Help page, there is no option to edit predefined help. You can duplicate a predefined help file, edit the copy, and optionally inactivate the original.

Where Predefined Help Appears

All predefined help comes with preassigned help locations, including at least one based on the hierarchy of type Business Processes. Many also have predefined
page or section values that indicate where the help can be accessed from help windows in the application.

To change where predefined help appears, either in the help site navigators or in the application, create a duplicate in the Manage Custom Help page. Change or add help locations to your custom copy, and inactivate the original.

**Customizing PDF Guides: Worked Example**

This example demonstrates how to customize a PDF guide that came with your local installation of Oracle Fusion Applications Help. This guide is currently not available from any help window in the application.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What changes do you need to make to the guide?</td>
<td>Change the title of a chapter and remove a section in that chapter, to hide content about a particular subject</td>
</tr>
<tr>
<td>Which help window should the customized guide appear in?</td>
<td>The help window for the entire Welcome dashboard of Oracle Fusion Applications</td>
</tr>
<tr>
<td>Which help navigators should the customized guide appear in, and on which node?</td>
<td>Same as the original guide, plus the path associated with the help window</td>
</tr>
<tr>
<td>Do you want to limit access to the customized guide?</td>
<td>No, same as the original guide</td>
</tr>
</tbody>
</table>

Edit a copy of the original PDF guide, and use the Manage Custom Help page to replace the original PDF guide with your new file.

**Copying and Editing the PDF Guide**

1. Open the original PDF guide from the help site and save a copy to your desktop. Leave open the help file for the guide.
2. Using a PDF editor application, change the title of the chapter wherever the chapter title appears. Delete the content you want to hide from users.
3. Make sure that your new PDF guide is less than 6 MB.

**Replacing the Original PDF Guide**

1. In the help file that you still have open for the original PDF guide, click the **Edit** link.
2. On the Create Help page, use the default values except where indicated.
3. Update the title to the name that you want to display to users.
4. In the **File Name** field, browse for and select your customized guide.
5. Delete any keywords or parts of the description relevant to the content you removed from the PDF guide.
6. Add a help location with the Business Processes hierarchy type and select **Information Technology Management** as the level 1 node, **Manage Enterprise Application Software** as the level 2 node, and **Use Applications** as the level 3 node.
7. Select **Welcome page** in the **Page or Section** column.

8. Click **Save and Close**. The help file for the original PDF guide is automatically set to inactive.

### Adding Custom UPK Content to Help: Worked Example

This example demonstrates how to add custom Oracle User Productivity Kit (UPK) topics as demo help files. These help files function like any predefined help file for demos. You can search and include these files in help windows and navigators as you would other help.

In this scenario, you are adding two demos about social networking, to appear in help windows on the Welcome dashboard.

**Note**

Your demo must be made with UPK 3.6.1 or later to be added as help.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What UPK content do you want to add to help?</td>
<td>From a UPK module containing five topics, add two as custom demos on the help site</td>
</tr>
</tbody>
</table>
| Which help navigators should each demo appear in, and on which node? | Because the two demos are about social networking:  
  • Search by Common Tasks navigator, under the Collaboration node  
  • Search by Business Process navigator, under Information Technology Management - Manage Networking and Communications - Manage Social Networking Capabilities |
| Which help window should each demo appear in? | On the Welcome dashboard of Oracle Fusion Applications, one demo goes in the help window in the Activity Stream region, and the other in the People Connection region |
| Do you want to limit access to the help files for the demos? | No |

Generate a report of UPK document IDs, which you will use when creating custom help, to identify the UPK topics that you want to add. Publish the UPK module as a player package, then create custom help for the UPK topics that you want to use as help demos.

**Generating a UPK Document ID Report**

1. In the UPK Developer, select **Details View**.
2. Right-click any column header, for example Name, and select **Column Chooser**.
3. In the Column Chooser dialog box, click and drag the Document ID column header and drop it after the Name column. Close the Column Chooser dialog box.
4. From the File menu, select to print, and save the output as a Microsoft Excel file to your desktop.

Creating the Player Package

1. From the UPK Developer, make sure that the topics that you want to add as demos have the See It play mode. The topics can also have other modes, but only the See It mode is included in the custom help file.

2. Publish the module, specifying any location for the output and selecting to publish the selection only.

3. In the Formats section of the Publish Content window, select the **Player** check box under the **Deployment** check box group.

4. In the Player section, select the **Include HTML Web Site** check box, to ensure that the custom help file includes a text-only version of the UPK topic.

5. Finish the publishing process, after optionally setting other options.

6. Navigate to the location where you specified the output to be generated.

7. In the Publishing Content folder, copy the PlayerPackage folder and add it to the web server where you store UPK content.

Creating Custom Help for Demos

1. Open the help window in the Activity Stream region on the Welcome dashboard of Oracle Fusion Applications, and click **Manage Custom Help**.

2. On the Manage Custom Help page, the page or section and hierarchy values are populated with the values for the Activity Stream region.

3. Click **Create**.

4. On the Create Help page, complete the fields in the General Information section, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title</strong></td>
<td>The name of the UPK topic.</td>
</tr>
<tr>
<td><strong>Source Type</strong></td>
<td>Oracle User Productivity Kit</td>
</tr>
<tr>
<td><strong>File Location</strong></td>
<td>The full URL of the player package folder on the Web server, for example, http://&lt;your domain&gt;.com/UPKcontent/PlayerPackage.</td>
</tr>
<tr>
<td><strong>Document ID</strong></td>
<td>The document ID of the UPK topic to add to the help window in the Activity Stream region. You can copy and paste this ID from the Microsoft Excel file that you generated earlier.</td>
</tr>
<tr>
<td><strong>Help Type</strong></td>
<td>Demo</td>
</tr>
<tr>
<td><strong>Help Security Group</strong></td>
<td>Unsecured</td>
</tr>
<tr>
<td><strong>Keywords</strong></td>
<td>Terms relevant to the demo.</td>
</tr>
</tbody>
</table>
The Help Location section contains values for the help window in the Activity Stream region. This help file will also appear in the Search by Business Process navigator under this predefined hierarchy.

5. Click Save and Close.

6. On the Manage Custom Help page, open the help locations for the help file that you just created.

7. Add a help location with the Welcome hierarchy type and select Collaboration Features as the level 1 node.

8. Add another help location with the Business Processes hierarchy type and select Information Technology Management as the level 1 node, Manage Networking and Communications as the level 2 node, and Manage Social Networking Capabilities as the level 3 node.

9. Click Save and Close.

10. Starting at the Connections region, repeat steps 1 to 9 for the other UPK topic that you want to add.

**Customizing Embedded Help: Highlights**

You can customize help that is embedded in the application, for example hints and help windows, for all users of Oracle Fusion Applications.

Embedded help customization is fully described in the Oracle Fusion Applications Extensibility Guide.

- Edit, create, or delete hint text that appears on hover over buttons, links, icons, or tab titles.
  
  See: Customizing or Adding Bubble Embedded Help

- Edit, create, or delete other types of embedded help.
  
  See: Customizing or Adding Static Instructions, In-field Notes, Terminology Definitions, and Help Windows

**FAQs for Define Help Configuration**

**Can I determine which users have access to collaboration features that involve interaction with the public help site?**

No, you cannot determine access for select users, but you can disable access for all users of your local installation of Oracle Fusion Applications Help. Use the Set Help Options page to disable sharing ratings and comments with the public help site on the Internet, and disable displaying ratings from the public site. You should also deselect the Access to Internet-Based Help Features feature choice.

If you are using the public instead of a local installation, then all collaboration features are always available to any user with access to the Web.
When do I link to the Oracle User Productivity Kit library from the help site?

Provide a link to your Oracle User Productivity Kit (UPK) library if you have UPK licensed and custom UPK content to share with your users. You give them access to a library of custom UPK content in addition to any custom UPK demos that you added to the help site itself. UPK demos that you add as custom help are available only in the See It mode, so the library can include the same demo in other modes. If you have UPK versions earlier than 3.6.1, then you cannot add UPK demos as custom help, so the link is the only way for users to access custom UPK content from the help site.

How can I find the URL to the Oracle User Productivity Kit library?

The URL to enter on the Set Help Options page should be the full path from the Web server where you are hosting your Oracle User Productivity Kit (UPK) content to the index.html file that opens the table of contents for the library, for example, http://<your domain>.com/UPKcontent/PlayerPackage/index.html. In this example, you or your UPK administrator would publish one UPK player package that contains all the content to be linked to from Oracle Fusion Applications Help, as well as the index.html file, and place the PlayerPackage folder in a manually created folder called UPKcontent on the Web server.

Who can add and manage custom help?

Users with the Oracle Fusion Help Text Administration duty have access to customize help in a local installation of Oracle Fusion Applications Help. This duty is assigned by default to various job roles, in particular the administrators for product families. Assign the role to other users who need access to customize help.

How can I restrict help content to specific user roles?

When you create or edit help, select a help security group that represents the set of roles that you want to have access to the help. If you do not see the Security Group field, then your administrator has not selected the Custom Help Security feature choice. The Unsecured group has no associated roles, so anyone can view the help. The predefined Secured group includes all internal employees and contingent workers, unless this group has been edited. You can create security groups and associate roles using the Manage Help Security Groups page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Help Security Groups task. Your new security groups are immediately available for use to secure new or edited help files.

Why can't I select and add help to a location?

You must specify a page or section to add the existing help to. To ensure that help is added to the correct help window, go to the page or section in the application, click the Help icon, and click the Manage Custom Help link in the help window. Alternatively, in the Manage Custom Help page, search for at least a page or section and a level 1 value for the Business Processes hierarchy type before selecting the Select and Add option.
You cannot select and add help to a particular hierarchy, on the Manage Custom Help page, without a page or section. To add just a hierarchy, search for the help file, add a new help location, and specify only the hierarchy information.

**What happens to custom help when a help patch is applied?**

Oracle Fusion Applications Help patches update all help files, both active and inactive, except custom help. Custom help files are not affected by patches. Consider reviewing inactive files to see if you want to activate the updated version, or to make similar edits to the custom versions of those files, if any.

**Define Application Toolkit Configuration**

**Define Application Toolkit Configuration: Overview**

Oracle Fusion Application Toolkit (ATK) is an application that provides various core components of Oracle Fusion Applications, including the Welcome dashboard, Oracle Fusion Applications Help, the Reports and Analytics pane, and the Watchlist feature. Use the Define Application Toolkit Configuration task list to set up and maintain some of these components for all users, and the Define Help Configuration task list for Oracle Fusion Applications Help.

Use the Setup and Maintenance work area to access the tasks in the Define Application Toolkit Configuration task list.

**Watchlist Setup: Points to Consider**

For all users across the site, you can disable or enable predefined Watchlist categories and items, edit their names, and determine how often item counts refresh. You cannot delete predefined Watchlist categories and items, nor create any for the site. Users can create their own Watchlist items through saved searches.

Access the Set Watchlist Options page by starting in the Setup and Maintenance Overview page and searching for the Set Watchlist Options task.

**Disabling Predefined Categories and Items**

Use the Set Watchlist Options page to enable or disable predefined Watchlist categories and items. Disabling any category or item also disables associated processes involved in calculating the Watchlist item counts for all users. These processes include creating data caches, performing security checks, invoking services across domains, running queries, and so on.

An item with the **Predefined** type represents the actual predefined Watchlist item that appears in the Watchlist. If you disable this type of Watchlist item, then:

- The item is not available for users to display in their watchlist
- The item is removed from any watchlist where it is currently displayed

A Watchlist item with the **User-created saved search** type does not appear in the Watchlist; it controls the display of the **Manage Watchlist** button or menu item in pages with saved searches. If you disable this type of Watchlist item, then:
• The **Manage Watchlist** option is not available to users in the corresponding work area, so users cannot use their own saved searches as Watchlist items. A message is displayed to users when they try to use this option.

• Any user-defined saved searches from that work area already used as Watchlist items are no longer available in the users' watchlist. The user-defined saved searches are still available to be used for searching, but not for the Watchlist.

If you disable a Watchlist category, then the category is not available for users to include in their watchlist, and all Watchlist items within the category are also disabled.

**Note**

If a product is not installed, then corresponding roles should not be granted to users, so the categories for that product would not be displayed. Even if roles are granted, users would see only zero count items. To make sure that only appropriate categories are available, disable categories for all products that are not installed, and disable irrelevant items, if any, for products that are installed.

Ultimately, the Watchlist for any user contains the subset of categories and items that are enabled in the Set Watchlist Options page:

• Plus any items based on user-defined saved searches

• Minus any categories or items that the user chooses to hide using Watchlist preferences

• Minus any items with no results found, if the user chooses to hide such items using Watchlist preferences

**Specifying Refresh Intervals**

All Watchlist items have a predefined refresh interval, which controls how often the query that calculates the count for a Watchlist item can be run. Use the Set Watchlist Options page to edit the interval values. What you specify as the refresh interval for a Watchlist item of type User-created Saved Search applies to all Watchlist items based on saved searches created by users on the corresponding search page.

When the user is in the Welcome dashboard with the Watchlist open for at least two and a half minutes, the query automatically runs for all Watchlist items if no refresh already ran in this user session. To subsequently run the query again, users can manually refresh the Watchlist region. The **Refresh** icon is enabled after five minutes since the last refresh.

**Note**

During a refresh, the query runs for an individual Watchlist item only if the time since the last query for this item is equal to or greater than the specified refresh interval. Since the manual refresh of the entire Watchlist is not available until five minutes after the last refresh, you should not set a Watchlist item refresh interval that is less than five minutes.

When users open Watchlist from the global area, a refresh automatically runs if five minutes have passed since the last refresh. During this refresh, the query
runs for an individual Watchlist item only if the time since the last query for this item is equal to or greater than the specified refresh interval.

For example, you set the interval to eight minutes for a particular Watchlist item. When the user signs in and goes to the Welcome dashboard, with the Watchlist open, the query automatically runs for this Watchlist item after two and a half minutes. Every two and a half minutes after, a check is performed for stale counts and new cached counts are displayed.

Five minutes after the query ran, the Refresh icon is enabled and the user performs a manual refresh. However, the query does not run for this Watchlist item, because the refresh interval is eight minutes. The user navigates away from the Welcome dashboard and opens the Watchlist from the global area six minutes later. A refresh automatically runs because more than five minutes have passed since the last refresh. This time, the query runs for this Watchlist item because it has been more than eight minutes since the query last ran for this item.

**Editing Predefined Category and Item Names**

Predefined Watchlist category and item names are stored as meanings of standard lookups. Lookup types for predefined categories end with WATCHLIST, for example EXM_EXPENSES_WATCHLIST. Edit the lookup type meaning to change the category name. To change item names, edit lookup code meanings for that lookup type.

**FAQs for Define Application Toolkit Configuration**

**How can I set up the Reports and Analytics pane for all users?**

Click Edit Settings in the Reports and Analytics pane, or start in the Setup and Maintenance work area and search for the Map Reports to Work Areas task. If you do the former, then you see all the reports that are currently mapped, or included, in the Reports and Analytics pane on the work area you are in. If you do the latter, then select a work area to display all the reports that are currently mapped to that work area. Select and remove any currently mapped report, or add reports from the Oracle Business Intelligence (BI) Presentation catalog.

Any changes you make apply to all users with access to the mapped work area. You can, however, restrict access to specific reports for specific users, and this security is not limited to the Reports and Analytics pane. For more information on managing objects in the Oracle BI Presentation Catalog and setting permissions, see the Oracle Fusion Middleware User’s Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

Oracle Business Intelligence Publisher reports must be registered as processes to be submitted from the Reports and Analytics pane. For more information on registering custom reports with Oracle Enterprise Scheduler and making reports available to users in the Reports and Analytics pane, see the Oracle Fusion Applications Extensibility Guide.

**Why can’t I see reports when mapping reports to work areas for the Reports and Analytics pane?**

If you are using the Map Reports to Work Areas task from the Setup and Maintenance work area, then select a work area to see the currently mapped reports. It is also possible that there are no reports currently mapped to the
selected work area, or to the work area on which you clicked **Edit Settings** in the Reports and Analytics pane. Alternatively, reports could be mapped, but you do not have access to them due to security.

Similarly, in the list of all available reports from the catalog, you can see only the reports that you have access to. You can request to be granted a role that has access to the reports that you want to map, or another administrator or business user with access to those reports can be granted the Reports and Analytics Region Administration Duty to be able to map reports to work areas.

**Maintain Common Reference Objects**

**Maintain Common Reference Objects: Overview**

The Maintain Common Reference Objects task list contains Oracle Fusion Middleware Extensions for Applications (Applications Core) tasks that provide support for implementation of common behaviors, such as data security or reference data sets.

Use this task list to manage common reference objects that are defined centrally and shared across applications, in addition to those that are specific to Applications Core functionality.

**Note**

Offerings also include application-specific tasks for managing Applications Core objects. For example, the Financials offering includes tasks such as Manage Receivables Descriptive Flexfields, and Manage Receivables Lookups.

For more information on configuring custom objects, see the Oracle Fusion Applications Extensibility Guide.

Use the Maintain Common Reference Objects task list, which you can access by starting in the Setup and Maintenance Overview page and searching for common reference object task lists.

**Define Application Taxonomy**

**Application Taxonomy: Highlights**

Application taxonomy is the organization of Oracle application components and functions in a hierarchical structure, from product lines to logical business areas. This hierarchy represents a breakdown of products into units based on how applications are installed and supported. Maintain this hierarchy on the Manage Taxonomy Hierarchy page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Taxonomy Hierarchy task.

A detailed introduction to application taxonomy is provided in the Oracle Fusion Applications Developer’s Guide.

**Hierarchy**

- The application taxonomy hierarchy contains various levels and types of nodes, or modules.
Usage

- Use application taxonomy to understand relationships among applications and between an application and its files. This information is helpful in managing various phases of the product lifecycle.

See: How to Manage the Lifecycle

Modules in Application Taxonomy: Explained

A module is any node in the application taxonomy hierarchy. The top level of the hierarchy is product line, followed by product family, application, and logical business area. There can be multiple levels of logical business areas, with one or more nested within a parent logical business area.

Product Line

A product line is a collection of products under a single brand name, for example, Oracle Fusion.

Product Family

A product family is a collection of products associated with a functional area that may or may not be licensed together as a single unit, for example Financials.

Application

An application is a single product within a product family, containing closely related features for a specific business solution, for example General Ledger.

Logical Business Area

A logical business area is a collection of business object definitions organized into a logical grouping. It contains the model objects, services, and UI components for those business objects. Logical business areas have their own hierarchy levels and in some cases can be two or three levels deep. Each leaf node has at least one business object and service, up to a maximum of four business objects and associated services. A logical business area with more than four business objects are further refined with child logical business area levels. Each of these parent-child levels is represented by a directory in the physical package hierarchy.

Managing Modules in Application Taxonomy: Points to Consider

Manage modules on the Create Child Module or Edit Module page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Taxonomy Hierarchy task. When you create a module, it is a child of the currently selected node in the application taxonomy hierarchy. This determines which values are available, for example for module type. Once
created, you cannot delete the module or move it elsewhere in the hierarchy.
As you create or edit modules, consider the following points regarding specific fields.

**Identifiers**

Module ID is the unique primary key for nodes in the taxonomy table. When you create a module, an ID is automatically generated. Once the module is created, you cannot update the ID.

Module key and alternative ID are additional identifiers of the module, presented in a way that is easier to read than the module ID. The module key is a string identifier, for example AP for the Oracle Fusion Payables application. The alternative ID is a numeric identifier, for example 1 for the Oracle Fusion product line. These IDs are provided for the product line, product family, and application modules, but you can optionally add them for logical business areas and new custom modules.

**Note**

Do not change the module key or alternative ID for predefined modules.

The product code is relevant only to application and logical business area modules. You can leave the field blank for other module types. The product code for applications is the short name that can be displayed in lists of application values, for example FND for Oracle Fusion Middleware Extensions for Applications.

**Names**

Module name is the logical name for the module and is always available. The name must be unique among nodes in the same hierarchy level with the same parent, but try to make it as unique in the whole hierarchy as possible.

The user name and description can appear to users in other parts of Oracle Fusion Applications, so make sure that the values are something that users know to represent the module.

**Usage Types**

Though you can update the usage type to reflect the current state of the module, just doing so does not affect the actual state. For example, setting a module as installed does not mean it is actually installed if the installation itself has not taken place. Installation refers to operations related to laying down all the components needed to create an Oracle Fusion Applications environment, while deployment is the process that starts the managed servers and clusters and facilitates the actual use of product offerings. A licensed module is available for installation and deployment, and a deployed module is considered actively used when actually used by users.

**Seed Data**

If seed data is allowed, then seed data such as flexfields and lookups can be extracted for the module using seed data loaders. By default, extract is allowed for all predefined modules of type application and logical business area.
Associations

You can associate a logical domain to modules of type product family, as well as one or more enterprise applications to modules of type application. This association represents the relationship between the taxonomy modules and the corresponding domain and enterprise applications stored in the Oracle Fusion Applications Functional Core (ASK) tables.

Define Reference Data Sharing

Reference Data Sharing: Explained

Reference data sharing facilitates sharing of configuration data such as jobs and payment terms, across organizational divisions or business units. You define reference data sets and determine how the data is shared or partitioned. Use reference data sets to reduce duplication and maintenance by sharing common data across business entities where appropriate. Depending on the requirement (specific or common), each business unit can maintain its data at a central location, using a set of values either specific to it or shared by other business units.

You can share reference data after it is filtered on the basis of sets. A common reference data set is available as the default set, which can be assigned to several business units sharing the same reference data. For commonly used data such as currencies, you can use the common reference data set and assign it to multiple business units in various countries that use the same currency. In cases where the default set cannot be assigned to an entity, you can create specific sets. The data set visible on the transactional page depends on the sharing method used to share reference data.

For example, XYZ Corporation uses the same grades throughout the entire organization. Instead of managers in different business units setting up the same grades, XYZ Corporation decides to create a set called Grades and assign the grades reference data group for all business units in the organization to the Grades set, so that the grades can be shared.

Note

For specific information on configuring reference data sharing for a particular object or product, refer to its product documentation.

Reference Data Sets: Explained

Reference data sets are logical groups of reference data that can be accessed by various transactional entities depending on the business context. Oracle Fusion Applications contains a common reference data set as well as an enterprise set that may be used as a default set. Depending on your business requirement you can create and maintain additional reference data sets, while continuing to use the common reference data set.

Consider the following scenario.
Your enterprise can decide that some aspects of corporate policy should affect all business units and leave other aspects to the discretion of the business unit manager. This allows your enterprise to balance autonomy and control for each business unit. For example, if your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level, you can let managers define their own sales methods, but define payment terms centrally. In this case, each business unit would have its own reference data set for sales methods, and there would be one central reference data set for payment terms assigned to all business units.

**Partitioning**

The partitioning of reference data and creation of data sets enable you to create reference entities across tables or lookup types, and share modular information and data processing options among business units. With the help of partitioning, you can choose to create separate sets and subsets for each business unit depending upon its business requirement, or create common sets or subsets to enable sharing reference data between several business units, without the need for duplicating the reference data. Partitioning provides you the flexibility to handle the reference data in a way appropriate to your business needs.

The following figure illustrates the reference data sharing method (assignment to one set only, with common values) where the user can access the data assigned to a specific set in a particular business unit, as well as access the data assigned to the common set.

**Reference Data Sets and Sharing Methods: Explained**

Oracle Fusion Applications reference data sharing feature is also known as SetID. The reference data sharing functionality supports operations in multiple ledgers, business units, and warehouses, thereby reducing the administrative
burden and decreasing the time needed to implement new business units. For example, you can share sales methods, transaction types, or payment terms across business units or selected other data across asset books, cost organizations, or project units.

The reference data sharing features use reference data sets to which reference data is assigned. The reference data sets group assigned reference data. The sets can be understood as buckets of reference data assigned to multiple business units or other application components.

**Reference Data Sets**

You begin this part of your implementation by creating and assigning reference data to sets. Make changes carefully as changes to a particular set will affect all business units or application components using that set. You can assign a separate set to each business unit for the type of object that is being shared. For example, assign separate sets for payment terms, transaction types, and sales methods to your business units.

Your enterprise can decide that some aspects of corporate policy should affect all business units and leave other aspects to the discretion of the business unit manager. This allows your enterprise to balance autonomy and control for each business unit. For example, if your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level, you can let managers define their own sales methods, but define payment terms centrally. In this case, each business unit would have its own reference data set for sales methods, and there would be one central reference data set for payment terms assigned to all business units.

The reference data sharing is especially valuable for lowering the cost of setting up new business units. For example, your enterprise operates in the hospitality industry. You are adding a new business unit to track your new spa services. The hospitality divisional reference data set can be assigned to the new business unit to quickly setup data for this entity component. You can establish other business unit reference data in a business unit specific reference data set as needed.

**Reference Data Sharing Methods**

There are variations in the methods used to share data in reference data sets across different types of objects. The following list identifies the methods:

- Assignment to one set only, no common values allowed. The simplest form of sharing reference data that allows assigning a reference data object instance to one and only one set. For example, Asset Prorate Conventions are defined and assigned to only one reference data set. This set can be shared across multiple asset books, but all the values are contained only in this one set.

- Assignment to one set only, with common values. The most commonly used method of sharing reference data that allows defining reference data object instance across all sets. For example, Receivables Transaction Types are assigned to a common set that is available to all the business units without the need to be explicitly assigned the transaction types to each business unit. In addition, you can assign a business unit specific set of transaction types. At transaction entry, the list of values for transaction types includes transaction types from the set assigned to the business unit,
as well as transaction types assigned to the common set that is shared across all business units.

- Assignment to multiple sets, no common values allowed. The method of sharing reference data that allows a reference data object instance to be assigned to multiple sets. For instance, Payables Payment Terms use this method. It means that each payment term can be assigned to one or more than one set. For example, you assign the payment term Net 30 to several sets, but the payment term Net 15 is assigned to only your corporate business unit specific set. At transaction entry, the list of values for payment terms consists of only one set of data; the set that is assigned to the transaction’s business unit.

Note: Oracle Fusion Applications contains a reference data set called Enterprise. Define any reference data that affects your entire enterprise in this set.

**Assigning Reference Data Sets to Reference Objects: Points to Consider**

You can assign the reference data sets to reference objects on the Manage Reference Data Set Assignments page. For multiple assignments, you can classify different types of reference data sets into groups and assign them to reference entity objects. The assignment takes into consideration the determinant type, determinant, and reference group, if any.

**Determinant Types**

The partitioned reference data is shared based on a business context setting called the determinant type. It is the point of reference used in the data assignment process. The following table lists the determinant types used in the reference data assignment.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset Book</td>
<td>Information about the acquisition, depreciation, and retirement of an asset that belongs to a ledger or a business unit.</td>
</tr>
<tr>
<td>Business Unit</td>
<td>The departments or organizations within an enterprise.</td>
</tr>
<tr>
<td>Cost Organization</td>
<td>The organization used for cost accounting and reporting on various inventory and cost centers within an enterprise.</td>
</tr>
<tr>
<td>Project Unit</td>
<td>A logical organization within an enterprise that is responsible for enforcing consistent project management practices.</td>
</tr>
<tr>
<td>Reference Data Set</td>
<td>References to other shared reference data sets.</td>
</tr>
</tbody>
</table>

**Determinant**

The determinant or determinant value is the value that corresponds to the selected determinant type. The determinant is one of the criteria for selecting the appropriate reference data set. For example, when managing set assignments for the set determinant type, Reference Data Set is the determinant type, and you
would enter the corresponding set code value as the corresponding determinant value.

Reference Groups

A transactional entity may have multiple reference entities (generally considered to be setup data) that are treated in the same manner because of commonness in implementing business policies and legal rules. Such reference entities in your application are grouped into logical units called reference groups, based on the functional area and the partitioning requirements that they have in common. For example, all tables and views that define Sales Order Type details might be part of the same reference group.

Note

The reference groups are predefined in the reference groups table and are available for selection and assignment.

Product Specific Reference Data Sharing: Explained

Some products required special logic for reference data sharing and have implemented their own domain specific ways for sharing data.

Items and supplier sites are two such product specific reference data objects that use product specific mechanisms to share data.

Items

If you share your items across warehouses or manufacturing facilities, you can access them through a common item master. Configure one or multiple item masters for your enterprise, based your enterprise structure. A single item master is recommended because it provides simpler and more efficient maintenance. However, in rare cases, it may be beneficial to keep multiple item masters. For example, if you acquire another enterprise and need to continue to operate your lines of business separately, maintaining a second item master might be the best decision.

Suppliers Sites

You can approve particular suppliers to supply specified commodities and authorize your business units to buy from those suppliers when the need arises. For example, you might be a household cleaning products manufacturer and need dyes, plastics, and perfumes to make your products. You purchase from a central supplier 70% of your perfume supplies with an additional supplier, in reserve, from whom you purchase the remaining 30%. At the same time, each of your business units purchases plastics and dyes from the same supplier, but from different local supplier sites to save transportation costs.

To implement business unit specific supplier sites, Oracle Fusion Procurement supports a method for defining supplier sites as owned and managed by the business unit responsible for negotiating the supplier terms. Your other business units that have a service provider relationship defined with your procurement business unit, subscribe to the supplier sites using the supplier site assignments feature. In addition, Procurement allows sharing of the following procurement data objects across business units:
• Supplier qualification data, such as approved supplier lists
• Catalog content, such as agreements, smart forms, public shopping lists, and content zones
• Procurement configuration data

**FAQs for Define Reference Data Sharing**

**What reference data objects can be shared across business units?**

The following list contains the reference data objects for the Oracle Fusion Applications that can be shared across business units and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trading Community Model</td>
<td>Customer Account Relationship</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Trading Community Model</td>
<td>Customer Account Site</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Trading Community Model</td>
<td>Sales Person</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Opportunity Management</td>
<td>Sales Method Group</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Work Management</td>
<td>Assessment Templates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Enterprise Contracts</td>
<td>Contract Types</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Sales</td>
<td>Sales Method</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Common Components</td>
<td>Activity Templates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Payables</td>
<td>Payment Terms</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
<tr>
<td>Receivables</td>
<td>Accounting Rules</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Aging Buckets</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Auto Cash Rules</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Collectors</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Lockbox</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Memo Lines</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Payment Term</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Remit To Address</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Application Name</td>
<td>Reference Data Object</td>
<td>Method of Sharing</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Receivables</td>
<td>Revenue Contingencies</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Transaction Source</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Transaction Type</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Advanced Collections</td>
<td>Collections Setups</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Advanced Collections</td>
<td>Dunning Plans</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Tax</td>
<td>Tax Classification Codes</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Performance Management</td>
<td>Performance Templates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Departments</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Jobs</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Locations</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Grades</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Project Billing</td>
<td>Project and Contract Billing</td>
<td>Assignment to multiple sets, common values not allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Accounting Definition</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Rates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Distributed Order Orchestration</td>
<td>Hold Codes</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Distributed Order Orchestration</td>
<td>Orchestration Process</td>
<td>Assignment to one set only, with common values</td>
</tr>
</tbody>
</table>

**What reference data objects can be shared across asset books?**

The following list contains the reference data objects for Oracle Fusion Assets that can be shared across asset books and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets</td>
<td>Bonus Rules</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Depreciation Ceilings</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Depreciation Methods</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Assets</td>
<td>Asset Descriptions</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
</tbody>
</table>
What reference data objects can be shared across cost organizations?

The following list contains the reference data objects for Oracle Fusion Cost Management that can be shared across cost organizations and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost Management</td>
<td>Cost Structure</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
</tbody>
</table>

What reference data objects can be shared across project units?

The following list contains the reference data objects for Oracle Fusion Project Foundation that can be shared across project units and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Foundation</td>
<td>Project Definition</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Transaction Types</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
</tbody>
</table>

Define Lookups

Lookups: Explained

Lookups are lists of values in applications. You define a list of values as a lookup type consisting of a set of lookup codes, each code’s translated meaning, and optionally a tag. End users see the list of translated meanings as the available values for an object.

Lookups provide a means of validation and lists of values where valid values appear on a list with no duplicate values. For example, an application might store the values Y and N in a column in a table, but when displaying those values in the user interface, Yes or No (or their translated equivalents) should be available for end users to select. For example, the two lookup codes Y and N are defined in the REQUIRED_INDICATOR lookup type.
In another example, a lookup type for marital status has lookup codes for users to specify married, single, or available legal partnerships.

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Lookup Code</th>
<th>Meaning</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAR_STATUS</td>
<td>M</td>
<td>Married</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Single</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Registered Partner</td>
<td>+NL</td>
</tr>
<tr>
<td></td>
<td>DP</td>
<td>Domestic Partner</td>
<td>-FR, AU</td>
</tr>
</tbody>
</table>

In this case, tags are used for localizing the codes. All legislations list Married and Single. Only the Dutch legislation lists Registered Partner. And all legislations except France and Australia also list Domestic Partner.

When managing lookups, you need to understand the following.

- Using lookups in applications
- Customization levels
- Accessing lookups
- Enabling lookups
- The three kinds of lookups: standard, common, and set enabled

**Using Lookups in Applications**

Use lookups to provide validation or a list of values for a user input field in a user interface.

An example of a lookup used for validation is a flexfield segment using a table-validated value set with values from a lookup type. An example of a lookup in a list of values is a profile option’s available values from which users select one to set the profile option. Invoice Approval Status gives the option of including payables invoices of different approval statuses in a report. The lookup code values include All so that users can report by all statuses: Approved, Resubmitted for approval, Pending or rejected, and Rejected.

**Customization Level**

The customization level of a lookup type determines whether the lookups in that lookup type can be edited. This applies data security to lookups.

Some lookup types are locked so no new codes and other changes can be added during implementation or later, as needed. Depending on the customization level of a lookup type, you may be able to change the codes or their meanings. Some lookups are designated as extensible, so new lookup codes can be created during implementation, but the meanings of predefined lookup codes cannot be modified. Some predefined lookup codes can be changed during implementation or later, as needed.

The customization levels are user, extensible, and system. The following table shows which lookup management tasks are allowed at each customization level.
<table>
<thead>
<tr>
<th>Allowed Task</th>
<th>User</th>
<th>Extensible</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deleting a lookup type</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Inserting new codes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Updating start date, end date, and enabled fields</td>
<td>Yes</td>
<td>Yes, only if the code is not predefined data</td>
<td>No</td>
</tr>
<tr>
<td>Deleting codes</td>
<td>Yes</td>
<td>Yes, only if the code is not predefined data</td>
<td>No</td>
</tr>
<tr>
<td>Updating tags</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Updating module</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Predefined data means LAST_UPDATED_BY = SEED_DATA_FROM_APPLICATION.

If a product depends on a lookup, the customization level should be system or extensible to prevent deletion.

**Standard, Common, and Set-Enabled Lookups**

The available kinds of lookups are as follows.

<table>
<thead>
<tr>
<th>Lookup</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>Lists the available codes and translated meanings</td>
</tr>
<tr>
<td>Set enabled</td>
<td>Additionally associates a reference data set with the lookup codes</td>
</tr>
<tr>
<td>Common</td>
<td>Legacy lookups</td>
</tr>
</tbody>
</table>

Standard lookups are the simplest form of lookup types consisting only of codes and their translated meaning. They differ from common lookups only in being defined in the standard lookup view.

Common lookups exist for reasons of backward compatibility and differ from standard lookups only in being defined in the common lookup view.

Set enabled lookup types store lookup codes that are enabled for reference data sharing. At runtime, a set-enabled lookup code is visible because the value of the determinant identifies a reference data set in which the lookup code is present.

**Accessing Lookups**

Standard, set-enabled, and common lookups are defined in the Standard, Set-enabled, and Common views, respectively. Applications development may define lookups in an application view to restrict the UI pages where they may appear.

In lookups management tasks, lookups may be associated with a module in the application taxonomy to provide a criteria for narrowing a search or limiting the number of lookups accessed by a product specific task such as Manage Purchasing Lookups.
Enabling Lookups

A lookup type is reusable for attributes stored in multiple tables.

Enable lookups based on the following.

- Selecting an Enabled check box
- Specifying an enabled start date, end date, or both
- Specifying a reference data set determinant

If you make changes to a lookup, users must sign out and back in before the changes take effect. When defining a list of values for display rather than validation, limit the number of enabled lookup codes to a usable length.

Managing a Standard Lookup: Example

Creating a new standard lookup involves creating or selecting a lookup type to which the lookup code belongs, and determining appropriate values for the lookup codes and their meanings.

Note

You can only create or edit the lookup codes for a particular lookup type if its customization level supports it.

Creating a Lookup Type Called COLORS

Your enterprise needs a list of values for status to be used on various objects such as processes or users. The lookups are colors, so the lookup type you create is COLORS.

<table>
<thead>
<tr>
<th>Lookup type parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup type name</td>
<td>COLORS</td>
</tr>
<tr>
<td>Meaning</td>
<td>Status</td>
</tr>
<tr>
<td>Description</td>
<td>Status by color</td>
</tr>
<tr>
<td>Module</td>
<td>Oracle Fusion Middleware Extensions for Applications</td>
</tr>
</tbody>
</table>

The lookup codes you define for the COLORS lookup type are, BLUE, RED, GREEN, and YELLOW.

<table>
<thead>
<tr>
<th>Lookup Code</th>
<th>Meaning</th>
<th>Enabled</th>
<th>Display Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE</td>
<td>Urgent</td>
<td>No</td>
<td>4</td>
</tr>
<tr>
<td>RED</td>
<td>Stop</td>
<td>Yes</td>
<td>1</td>
</tr>
<tr>
<td>GREEN</td>
<td>Go</td>
<td>Yes</td>
<td>3</td>
</tr>
<tr>
<td>YELLOW</td>
<td>Caution</td>
<td>Yes</td>
<td>2</td>
</tr>
</tbody>
</table>
Understanding the Resulting Data Entry List of Values

Users need to respond to a process question by indicating whether to stop it, use caution, go ahead, or complete it urgently.

The list of values for the COLORS lookup type includes the meanings for the enabled codes.

<table>
<thead>
<tr>
<th>Displayed Value</th>
<th>Hidden ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop</td>
<td>RED</td>
</tr>
<tr>
<td>Caution</td>
<td>YELLOW</td>
</tr>
<tr>
<td>Go</td>
<td>GREEN</td>
</tr>
</tbody>
</table>

Analysis

The BLUE lookup code was not enabled and does not appear in the list of values. The display sequence of values in the list of values is alphabetical unless you enter a number manually to determine the order of appearance. Number 1 indicates the value listed first in the list of values.

Note

Only lookups that are enabled and active, meaning between start and end dates, are visible.

Understanding the Transaction Table

When users enter one of the values from the list of values for the lookup type COLORS, the transaction table records the lookup code. In this example, the code is stored in the Status column.

<table>
<thead>
<tr>
<th>Transaction number</th>
<th>User name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jane</td>
<td>RED</td>
</tr>
<tr>
<td>2</td>
<td>Bob</td>
<td>YELLOW</td>
</tr>
<tr>
<td>3</td>
<td>Alice</td>
<td>BLUE</td>
</tr>
</tbody>
</table>

The status for one user is BLUE because at the time they entered a value, BLUE was enabled. Disabling a lookup code does not affect transaction records in which that code is stored. Data querying and reporting have access to disabled lookup codes in transaction tables.

Managing Set-Enabled Lookups: Examples

Creating a new set-enabled lookup is similar to creating a standard lookup with the addition of specifying a reference data set determinant for the lookup codes.

Note

You can only create or edit the lookup codes for a particular lookup type if its customization level supports it.
The reference data set for a set-enabled lookup code is part of its foreign key. This is unlike other set-enabled entities.

Selecting a Reference Group for a Set-Enabled Lookup Type

By specifying a reference group for a set-enabled lookup type you indicate which reference data set assignments are available for its lookup codes. For example a COLORS lookup type might be set enabled for a Countries reference group that includes the US and EU reference data set assignments.

Selecting a Reference Data Set for a Set-Enabled Lookup

The reference data set determines which lookup code is included in the list of values. If a COLORS lookup type contains a RED, YELLOW, ORANGE, and GREEN lookup code, you can enable one RED lookup as coming from the US reference data set and another RED lookup as coming from the EU reference data set with divergent meanings.

<table>
<thead>
<tr>
<th>Reference Data Set</th>
<th>Lookup Code</th>
<th>Lookup Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>RED</td>
<td>Red</td>
</tr>
<tr>
<td>US</td>
<td>YELLOW</td>
<td>Yellow</td>
</tr>
<tr>
<td>US</td>
<td>GREEN</td>
<td>Green</td>
</tr>
<tr>
<td>EU</td>
<td>RED</td>
<td>Rouge</td>
</tr>
<tr>
<td>EU</td>
<td>ORANGE</td>
<td>Orange</td>
</tr>
</tbody>
</table>

In addition to divergent meanings for lookup codes based on associated reference data set, some lookup codes may be unique to one or another reference data set as the ORANGE lookup is to the EU reference data set in this example.

In another example, a lookup type called HOLD_REASON provides a list of reasons for applying a hold to a contract renewal. Reference data sets determine which codes are included in the hold reason list of values.

<table>
<thead>
<tr>
<th>Reference Data Set</th>
<th>Lookup Code</th>
<th>Lookup Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>SEC</td>
<td>SEC Compliance Review</td>
</tr>
<tr>
<td>US</td>
<td>DIR</td>
<td>Needs Director’s Approval</td>
</tr>
<tr>
<td>US</td>
<td>VP</td>
<td>Needs Vice President’s Approval</td>
</tr>
<tr>
<td>CHINA</td>
<td>CSRC</td>
<td>Pending China Securities Regulatory Commission Review</td>
</tr>
<tr>
<td>CHINA</td>
<td>PR</td>
<td>Needs President’s Approval</td>
</tr>
<tr>
<td>COMMON</td>
<td>REQUESTED</td>
<td>Customer Request</td>
</tr>
</tbody>
</table>

Using the Manage Set Assignments task, you have defined assignments that designate the China business unit to refer to the CHINA and the US business unit to refer to the US and all business units to refer to the COMMON set. When end users place a contract hold in the US business unit, only the three reason codes in US_SET are available. When placing a contract hold in the China business, only the two codes in China_SET are available.
FAQs for Define Lookups

How can I edit lookups?

You can edit the existing lookup codes of a lookup type or add new lookup codes on the Define Lookups pages, which you can access by starting in the Setup and Maintenance work area and searching for lookup tasks. You can edit the existing lookup codes of a lookup type, or add new lookup codes to a lookup type, if the customization level for the lookup type supports editing.

Why can't I see my lookup types?

Lookups are listed by lookup type. Typically lookup types are managed using tasks that handle a group of related lookups, such as Manage Geography Lookups. Each task gives you access only to certain lookup types. The generic tasks provide access to all lookup types of a kind, such as all common lookups using the Manage Common Lookups task.

If existing lookups are not available to the tasks of the Define Lookups activity, they may be validated for use in a lookup view that is not central to all applications or whose owning application has not been specified in a lookup view.

Lookups can only be managed in the Define Lookups tasks if the lookup’s view application is the standard lookups view, common lookups view, or set-enabled lookups view. Lookups defined in an application view can only be managed by following instructions provided by the owning application.

Note

A lookup type and its codes can only be defined in one lookup view.

What’s the difference between a lookup type and a value set?

A lookup type consists of lookup codes that are the values in a static list of values. Lookup code validation is a one to one match.

A table-validated value set can consist of values that are validated through a SQL statement, which allows the list of values to be dynamic.

Tip

A table validated value set can be defined based on any table, including the lookups table. This allows a lookup type to be made into a table-validated value set that can be used in flexfields.

<table>
<thead>
<tr>
<th>Area of Difference</th>
<th>Lookup Type</th>
<th>Value Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of values</td>
<td>Static</td>
<td>Dynamic if Table validation type</td>
</tr>
<tr>
<td>Validation of values</td>
<td>One to one match of meaning to code included in a lookup view, or through the determinant of a reference data set</td>
<td>By format or inclusion in a table</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Format type of values</td>
<td>char</td>
<td>varchar2, number, and so on</td>
</tr>
<tr>
<td>Length of value</td>
<td>Text string up to 30 characters</td>
<td>Any type of variable length from 1 to 4000</td>
</tr>
<tr>
<td>Duplication of values</td>
<td>Never. Values are unique.</td>
<td>Duplicate values allowed</td>
</tr>
<tr>
<td>Management</td>
<td>Managed by both administrators and end-users, except system lookups or predefined lookups at the system customization level, which cannot be modified.</td>
<td>Maintained by administrators, except some product flexfield codes, such as GL for Oracle Fusion General Ledger, which are maintained by end users</td>
</tr>
</tbody>
</table>

A lookup type cannot make use of a value from a value set.

Value sets can make use of standard, common, or set-enabled lookups.

Both lookup types and value sets are used to create lists of values from which users select values.

**What's a lookup tag used for?**

Tags on lookup codes allow you to add a label to your lookup codes.

Lookup tags are unvalidated and uninterpreted by lookups. A tag can be used to categorize lookups based on facilitating searches or guiding how a lookup should be used.

Document what the tag on a lookup represents and how to use it.

**Manage Messages**

**Messages: Highlights**

The message dictionary contains messages that tell users about business rule errors, such as missing or incorrect data, and how to resolve them, to warn users about the consequences of intended actions, and provide information in log files. These messages are defined for specific applications and modules, but a few are common messages that can be used in any application. All applications also use messages stored outside of the message dictionary.


**Managing Messages**

- Use the Manage Messages page to create and edit custom messages in the message dictionary, as well as edit predefined messages. Do not delete
predefined messages unless you are sure that they are not used anywhere. Refer to the Oracle Fusion Applications Developer’s Guide.

See: Introduction to Message Dictionary Messages

• Messages outside of the message dictionary, for example confirmations and field validations, are managed in Oracle Application Development Framework Faces components or through message resource bundles used for translation. Refer to the Oracle Fusion Middleware Web User Interface Developer’s Guide for Oracle Application Development Framework.

See: Displaying Hints and Error Messages for Validation and Conversion

See: Internationalizing and Localizing Pages

Creating and Editing Messages: Highlights

Each message in the message dictionary has many attributes and components, including message properties, text, and tokens, that you define when creating or editing the message.

Details about these messages are described in the Oracle Fusion Applications Developer’s Guide.

Message Properties

• The message type identifies the type of information that the message contains.

See: Understanding Message Types

• The message name and number are identifiers for the message. There are specific message number ranges for predefined messages in each application, and you should not edit numbers assigned to predefined messages. When creating custom messages, use only message numbers within the 10,000,000 to 10,999,999 range.

See: About Message Names

See: About Message Numbers

• The translation notes for predefined messages might contain internal content that you can disregard.

See: About Translation Notes

• The message category, severity, and logging enabled option are related to the incident and logging process.

See: About Grouping Messages by Category and Severity

See: Understanding Incidents and Diagnostic Logs with Message Dictionary

Message Text and Tokens

• The message text is comprised of various components, some of which are displayed only to select users. The short text is also referred to as the
message text, because all messages must have this component, but may or may not include any of the other components.

See: About Message Components

- Tokens are variables that represent values to be displayed in the message text.

See: About Tokens

**Common Messages: Points to Consider**

Common messages, which have message names that begin with FND_CMN and message numbers between 0 and 999, are used throughout Oracle Fusion Applications. Each common message can appear in multiple places in any product family. For example, the FND_CMN_NEW_SRCH message can be used for any search to indicate that no results were found. Common messages that are of type error or warning are part of the message dictionary.

**Editing Common Messages**

Because a common message can be used in any application, consider the ramifications if you edit any aspect of the message, including incident and logging settings. Changes would be reflected in all instances where the message is used. For example, if you change the message text, make sure that the text would make sense to all users across Oracle Fusion Applications who might see it.

**Creating Common Messages**

You can create custom common messages for use in multiple places within a single product. Do not begin the message name with FND_CMN, but use another suitable convention. The message number should be within the range that is designated for the product.

**FAQs for Manage Messages**

**How can I determine who sees the user or administrator components of a message?**

Set the Message Mode profile option to determine if the administrator or user details and action components of the message text is displayed. For example, you can set the profile option at the user level for a particular user to see the administrator information. All users can see the message short text and cause; the profile option does not apply to those message components. This profile option also applies only to messages in the message dictionary.

**Define ISO Reference Data**

**Natural Languages: Points to Consider**

Natural languages are all the languages that humans use, written and spoken. If a language is enabled, then users can associate it with entities, for example as
languages spoken by sales representatives. When managing natural languages, consider tasks to perform and best practices for entering particular values.

Tasks

Once you add a language, it cannot be deleted, just disabled. You can optionally associate natural languages with International Organization for Standardization (ISO) languages and territories, just for reference.

Values

When you create a natural language, use the alpha-2 ISO code as the language code, or, if not available, then alpha-3. If the language is not an ISO language, then use x- as a prefix for the code, for example x-ja for a Japanese dialect. Use the sgn code of ISO-639-2 for sign languages, followed by territory code, for example sgn-US for American Sign Language. You can also use Internet Assigned Numbers Authority (IANA) language tags.

The natural language description should be the language name with territory name in parenthesis where needed, for example English (Australia) and English (Canada).

Defining Currencies: Points to Consider

When creating or editing currencies, consider these points relevant to entering the currency code, date range, or symbol for the currency.

Currency Codes

You cannot change a currency code after you enable the currency, even if you later disable that currency.

Date Ranges

Users can enter transactions denominated in the currency only for the dates within the specified range. If you do not enter a start date, then the currency is valid immediately. If you do not enter an end date, then the currency is valid indefinitely.

Symbols

Even if you enter a symbol for a currency, the symbol is not always displayed when an amount is displayed in this currency. Some applications use currency symbols when displaying amounts. Others, like Oracle Fusion General Ledger, do not.

Euro Currency Derivation: Explained

Use the Derivation Type, Derivation Factor, and Derivation Effective Date fields to define the relationship between the official currency (Euro) of the European Monetary Union (EMU) and the national currencies of EMU member states. For
each EMU currency, you define its Euro-to-EMU fixed conversion rate and the effective starting date.

**Note**

If you need to use a different currency code for Euro, you can disable the predefined Euro currency and create a new one.

**Derivation Type**

The **Euro currency** derivation type is used only for the Euro, and the **Euro derived** derivation type identifies national currencies of EMU member states. All other currencies do not have derivation types.

**Derivation Factor**

The derivation factor is the fixed conversion rate by which you multiply one Euro to derive the equivalent EMU currency amount. The Euro currency itself should not have a derivation factor.

**Derivation Effective Date**

The derivation effective date is the date on which the relationship between the EMU currency and the Euro begins.

**FAQs for Define ISO Reference Data**

**When do I create or edit territories?**

Edit territory descriptions to determine how they are displayed in lists of country values throughout Oracle Fusion Applications. The predefined territories are all countries from the International Organization for Standardization (ISO) 3166 standard. You usually would not edit territory names or codes.

Do not edit National Language Support (NLS) territory codes, which are identifiers used in the system, unless you need to change the association between ISO and system territory. You usually would not edit the default currency, which is the value that defaults in the **Currency** field in Oracle Fusion Applications user preferences after the user first selects a territory.

Create territories if new countries emerge and the system has not yet been patched with the latest ISO country values.

**When do I create or edit industries?**

Edit industry descriptions to determine how they are displayed in Oracle Fusion Applications. You usually would not edit industry names, which are from the North American Industry Classification System (NAICS). Enabled industries are mainly used in the context of customization, though these values can also appear in any application.

Create industries if you have particular ones you need, for example for customization, that are not included in the NAICS standard.
When do I associate industries with territories?

Optionally associate industries with territories to provide an industry in territory value, used for customization. For example, administrators can customize a page in one way for users within an industry in one country, and another way for users within the same industry in another country. The administrator would select the appropriate industry in territory value to set the customization context.

When do I create or enable currencies?

Create currencies to use, for example for reporting purposes, if they are not already provided. All currencies from the International Organization for Standardization (ISO) 4217 standard are provided.

Enable any currency other than USD for use in Oracle Fusion Applications, for example for displaying monetary amounts, assigning to sets of books, entering transactions, and recording balances. Only USD is enabled by default.

What’s the difference between precision, extended precision, and minimum accountable unit for a currency?

Precision is the number of digits to the right of the decimal point used in regular currency transactions. Extended precision is the number of digits to the right of the decimal point used in calculations for this currency, and it must be greater than or equal to the standard precision. For example, USD would have 2 for precision because amounts are transacted as such, for example $1.00. For calculations, for example adding USD amounts, you might want the application to be more precise than two decimal digits, and would enter an extended precision accordingly.

Note

Some applications use extended precision. Others, such as Oracle Fusion General Ledger, do not.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for the cent. This unit does not necessarily correspond to the precision for all currencies.

What’s a statistical unit currency type?

The statistical unit currency type is used only for the Statistical (STAT) currency. The Statistical currency is used to record statistics such as the number of items bought and sold. Statistical balances can be used directly in financial reports, allocation formulas, and other calculations.

When do I create or edit ISO languages?

You can edit the names and descriptions of International Organization for Standardization (ISO) languages to determine how they are displayed in lists of ISO language values in Oracle Fusion Applications. The ISO languages are
from the ISO 639 standard. If there were changes to the ISO standard and the system has not yet been patched with the latest ISO values, you can update the ISO alpha-2 code or add languages as needed.

**When do I edit languages?**

Installed languages automatically appear on the Manage Languages page, so you do not manually enter newly installed languages. This page contains all languages available for installation and translation in Oracle Fusion Applications. Each dialect is treated as a separate language. The language codes and names are values used by the system.

You generally would not edit any of the detailed fields unless you really need to and know what they are.

**When do I create or edit time zones?**

Though all standard time zones are provided, optionally enable only a subset for use in lists of time zone values in Oracle Fusion Applications. You can add time zones if new zones became standard and the system has not yet been patched with the latest values.

**Manage Data Security Policies**

**Data Security in the Security Reference Implementation: Explained**

The reference implementation contains a set of data security policies that can be inspected and confirmed to be suitable or a basis for further implementation using the Authorization Policy Manager (APM).

The security implementation of an enterprise is likely a subset of the reference implementation, with the enterprise specifics of duty roles, data security policies, and HCM security profiles provided by the enterprise.

The business objects registered as secure in the reference implementation are database tables and views.

Granting or revoking object entitlement to a particular user or group of users on an object instance or set of instances extends the base Oracle Fusion Applications security reference implementation without requiring customization of the applications that access the data.

**Data Security Policies in the Security Reference Implementation**

The data security policies in the reference implementation entitle the grantee (a role) to access instance sets of data based on SQL predicates in a WHERE clause.

---

**Tip**

When extending the reference implementation with additional data security policies, identify instance sets of data representing the business objects that need to be secured, rather than specific instances or all instances of the business objects.
Predefined data security policies are stored in the data security policy store, managed in the Authorization Policy Manager (APM), and described in the Oracle Fusion Applications Security Reference Manual for each offering. A data security policy for a duty role describes an entitlement granted to any job role that includes that duty role.

**Warning**

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

The reference implementation only enforces a portion of the data security policies in business intelligence that is considered most critical to risk management without negatively affecting performance. For performance reasons it is not practical to secure every level in every dimension. Your enterprise may have a different risk tolerance than assumed by the security reference implementation.

**HCM Security Profiles in the Security Reference Implementation**

The security reference implementation includes some predefined HCM security profiles for initial usability. For example, a predefined HCM security profile allows line managers to see the people that report to them.

The IT security manager uses HCM security profiles to define the sets of HCM data that can be accessed by the roles that are provisioned to users.

**Data Roles**

The security reference implementation includes no predefined data roles to ensure a fully secured initial Oracle Fusion Applications environment.

The security reference implementation includes data role templates that you can use to generate a set of data roles with entitlement to perform predefined business functions within data dimensions such as business unit. Oracle Fusion Payables invoicing and expense management are examples of predefined business functions. Accounts Payable Manager - US is a data role you might generate from a predefined data role template for payables invoicing if you set up a business unit called US.

HCM provides a mechanism for generating HCM related data roles.

**Data Security: Explained**

By default, users are denied access to all data.

Data security makes data available to users by the following means.

- Policies that define grants available through provisioned roles
- Policies defined in application code

You secure data by provisioning roles that provide the necessary access. Enterprise roles provide access to data through data security policies defined for the inherited application roles.
When setting up the enterprise with structures such as business units, data roles are automatically generated that inherit job roles based on data role templates. Data roles also can be generated based on HCM security profiles. Data role templates and HCM security profiles enable defining the instance sets specified in data security policies.

When you provision a job role to a user, the job role implicitly limits data access based on the data security policies of the inherited duty roles. When you provision a data role to a user, the data role explicitly limits the data access of the inherited job role to a dimension of data.

Data security consists of privileges conditionally granted to a role and used to control access to the data. A privilege is a single, real world action on a single business object. A data security policy is a grant of a set of privileges to a principal on an object or attribute group for a given condition. A grant authorizes a role, the grantee, to actions on a set of database resources. A database resource is an object, object instance, or object instance set. An entitlement is one or more allowable actions applied to a set of database resources.

Data is secured by the following means.

<table>
<thead>
<tr>
<th>Data security feature</th>
<th>Does what?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data security policy</td>
<td>Grants access to roles by means of entitlement</td>
</tr>
<tr>
<td>Role</td>
<td>Applies data security policies with conditions to users through role provisioning.</td>
</tr>
<tr>
<td>Data role template</td>
<td>Defines the data roles generated based on enterprise setup of data dimensions such as business unit.</td>
</tr>
<tr>
<td>HCM security profile</td>
<td>Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code</td>
</tr>
<tr>
<td>Masking</td>
<td>Hides private data on non-production database instances</td>
</tr>
<tr>
<td>Encryption</td>
<td>Scrambles data to prevent users without decryption authorization from reading secured data</td>
</tr>
</tbody>
</table>

The sets of data that a user can access via roles are defined in Oracle Fusion Data Security. Oracle Fusion Data Security integrates with Oracle Platform Security Services (OPSS) to entitle users or roles (which are stored externally) with access to data. Users are granted access through the entitlement assigned to the roles or role hierarchy with which the user is provisioned. Conditions are WHERE clauses that specify access within a particular dimension, such as by business unit to which the user is authorized.

**Data Security Policies**

Data security policies articulate the security requirement "Who can do What on Which set of data," where 'Which set of data' is an entire object or an object instance or object instance set and 'What' is the object entitlement.

For example, accounts payable managers can view AP disbursements for their business unit.
<table>
<thead>
<tr>
<th>Who</th>
<th>can do</th>
<th>what</th>
<th>on which set of data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounts payable managers</td>
<td>view</td>
<td>AP disbursements</td>
<td>for their business unit</td>
</tr>
</tbody>
</table>

A data security policy is a statement in a natural language, such as English, that typically defines the grant by which a role secures business objects. The grant records the following:

- Table or view
- Entitlement (actions expressed by privileges)
- Instance set (data identified by the condition)

For example, disbursement is a business object that an accounts payable manager can manage by payment function for any employee expenses in the payment process.

**Note**

Some data security policies are not defined as grants but directly in applications code. The security reference manuals for Oracle Fusion Applications offerings differentiate between data security policies that define a grant and data security policies defined in Oracle Fusion applications code.

A business object participating in a data security policy is the database resource of the policy.

Data security policies that use job or duty roles refer to data security entitlement. For example, the data security policy for the Accounts Payable Manager job role refers to the view action on AP disbursements as the data security entitlement.

**Important**

The duty roles inherited by the job role can be moved and job roles reassembled without having to modify the data security.

As a security guideline, data security policies based on user session context should entitle a duty role. This keeps both function and data security policies at the duty role level, thus reducing errors.

For example, a Sales Party Management Duty can update Sales Party where the provisioned user is a member of the territory associated with the sales account. Or the Sales Party Management Duty can update Sales Party where the provisioned user is in the management chain of a resource who is on the sales account team with edit access. Or the Participant Interaction Management Duty can view an Interaction where the provisioned user is a participant of the Interaction.

For example, the Disbursement Process Management Duty role includes entitlement to build documents payable into payments. The Accounts Payable Manager job role inherits the Disbursement Process Management Duty role. Data security policies for the Disbursement Process Management Duty role authorize access to data associated with business objects such as AP disbursements within a business unit. As a result, the user provisioned with the Accounts Payable
Manager job role is authorized to view AP disbursements within their business unit.

A data security policy identifies the entitlement (the actions that can be made on logical business objects or dashboards), the roles that can perform those actions, and the conditions that limit access. Conditions are readable WHERE clauses. The WHERE clause is defined in the data as an instance set and this is then referenced on a grant that also records the table name and required entitlement.

**Data Roles**

Data roles are implemented as job roles for a defined set of data.

A data role defines a dimension of data within which a job is performed. The data role inherits the job role that describes the job. For example, a data role entitles a user to perform a job in a business unit.

The data role inherits abstract or job roles and is granted data security privileges. Data roles carry the function security privileges inherited from job roles and also the data security privilege granted on database objects and table rows.

For example, an accounts payables specialist in the US Business Unit may be assigned the data role Accounts Payables Specialist - US Business Unit. This data role inherits the job role Accounts Payables Specialist and grants access to transactions in the US Business Unit.

A data role may be granted entitlement over a set people.

For example, a Benefits Administrator A-E is allowed to administer benefits for all people that have a surname that begins with A-E.

Data roles are created using data role templates. You create and maintain data roles in the Authorization Policy Manager (APM). Use the Manage Data Roles and Security Profiles task to create and maintain HCM data roles in Oracle Fusion HCM.

**HCM Security Profiles**

HCM security profiles are used to secure HCM data, such as people and departments. You use HCM security profiles to generate grants for an enterprise role. The resulting data role with its role hierarchy and grants operates in the same way as any other data role.

For example, an HCM security profile identifies all employees in the Finance division.

Oracle Fusion Payroll uses HCM security profiles to secure project organizations. Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

**Masking and Encryption**

Oracle Fusion Applications uses masking to protect sensitive data from view by unauthorized users. Encryption APIs mask sensitive fields in applications user interfaces. Additionally, Oracle Data Masking is available for masking data in non-production instances and Oracle Transparent Data Encryption is available for protecting data in transit or in backups independent of managing encryption keys.
Database Resources and Data Security Policies: How They Work Together

A data security policy applies a condition and allowable actions to a database resource for a role. When that role is provisioned to a user, the user has access to data defined by the policy. In the case of the predefined security reference implementation, this role is always a duty role. Data roles generated to inherit the job role based on data role templates limit access to database resources in a particular dimension, such as the US business unit.

The database resource defines and instance of a data object. The data object is a table, view, or flexfield.

The following figure shows the database resource definition as the means by which a data security policy secures a data object. The database resource names the data object. The data security policy grants to a role access to that database resource based on the policy’s action and condition.

Database Resources

A database resource specifies access to a table, view, or flexfield that is secured by a data security policy.
• Name providing a means of identifying the database resource
• Data object to which the database resource points

Data Security Policies

Data security policies consist of actions and conditions for accessing all, some, or a single row of a database resource.

• Condition identifying the instance set of values in the data object
• Action specifying the type of access allowed on the available values

Note

If the data security policy needs to be less restrictive than any available database resource for a data object, define a new data security policy.

Actions

Actions correspond to privileges that entitle kinds of access to objects, such as view, edit, or delete. The actions allowed by a data security policy include all or a subset of the actions that exist for the database resource.

Conditions

A condition is either a SQL predicate or an XML filter. A condition expresses the values in the data object by a search operator or a relationship in a tree hierarchy. A SQL predicate, unlike an XML filter, is entered in a text field in the data security user interface pages and supports more complex filtering than an XML filter, such as nesting of conditions or sub queries. An XML filter, unlike a SQL predicate, is assembled from choices in the UI pages as an AND statement.

Tip

An XML filter can be effective in downstream processes such as business intelligence metrics. A SQL predicate cannot be used in downstream metrics.

Securing Data Access: Points to Consider

Oracle Fusion Applications supports securing data through role-based access control (RBAC) by the following methods.

<table>
<thead>
<tr>
<th>Method of securing data</th>
<th>Reason</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data roles apply explicit data security policies on job and abstract roles</td>
<td>Appropriate for job and abstract roles that should only access a subset of data, as defined by the data role template that generates the data role or by HCM security profiles.</td>
<td>Accounts Payable Manager - US data role to provide an accounts payable manager in the US business unit with access to invoices in the US business unit.</td>
</tr>
<tr>
<td>Data security policies</td>
<td>Define data access for application roles and provide inheriting job and abstract roles with implicit data security</td>
<td>Projects</td>
</tr>
</tbody>
</table>
If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

When a runtime session is created, Oracle Platform Security Services (OPSS) propagates only the necessary user to role mapping based on Oracle Fusion Data Security grants. A grant can specify entitlement to the following.

- Specific rows of data (data object) identified by primary key
- Groups of data (instance set) based on a predicate that names a particular parameter
- Data objects or instance sets based on runtime user session variables

Data is either identified by the primary key value of the row in the table where the data is stored. Or data is identified by a rule (SQL predicate) applied to the WHERE clause of a query against the table where the data is stored.

**Grants**

Oracle Fusion Data Security can be used to restrict the following.

- Rows that are returned by a given query based on the intended business operation
- Actions that are available for a given row

Grants control which data a user can access.

**Note**

Attribute level security using grants requires a data security policy to secure the attribute and the entitlement check enforces that policy.

A grant logically joins a user or role and an entitlement with a static or parameterized object instance set. For example, `REGION='WEST'` is a static object instance set and `REGION=GRANT_ALIAS.PARAMETER1` is a parameterized object instance set. In the context of a specific object instance, grants specify the allowable actions on the set of accessible object instances. In the database, grants are stored in FND_GRANTS and object instance sets are stored in FND_OBJECT_INSTANCE_SETS. Object access can be tested using the privilege check application programming interface (API).

**Securing a Business Object**

A business object is a logical entity that is typically implemented as a table or view, and corresponds to a physical database resource. The data security policies of the security reference implementation secure predefined database resources. Use the Manage Data Security Policies task to define and register other database resources.

Data security policies identify sets of data on the registered business object and the actions that may be performed on the business object by a role. The grant can be made by data instance, instance set or at a global level.
Note

Use parameterized object instance sets whenever feasible to reduce the number of predicates the database parses and the number of administrative intervention required as static object instances sets become obsolete. In HCM, security profiles generate the instance sets.

Data Role Templates: Explained

You use data role templates to generate data roles. You generate such data roles, and create and maintain data role templates in the Authorization Policy Manager (APM).

Note

HCM data roles are generated using the Manage Data Roles and Security Profiles task, which uses HCM security profiles, not data role templates, to define the data security condition.

The following attributes define a data role template.

- Template name
- Template description
- Template group ID
- Base roles
- Data dimension
- Data role naming rule
- Data security policies

The data role template specifies which base roles to combine with which dimension values for a set of data security policies. The base roles are the parent job or abstract roles of the data roles.

Note

Abstract, job, and data roles are enterprise roles in Oracle Fusion Applications. Oracle Fusion Middleware products such as Oracle Identity Manager (OIM) and Authorization Policy Manager (APM) refer to enterprise roles as external roles. Duty roles are implemented as application roles in APM and scoped to individual Oracle Fusion Applications.

The dimension expresses stripes of data, such as territorial or geographic information you use to partition enterprise data. For example, business units are a type of dimension, and the values picked up for that dimension by the data role template as it creates data roles are the business units defined for your
enterprise. The data role template constrains the generated data roles with grants of entitlement to access specific data resources with particular actions. The data role provides provisioned users with access to a dimensional subset of the data granted by a data security policy.

An example of a dimension is a business unit. An example of a dimension value is a specific business unit defined in your enterprise, such as US. An example of a data security policy is a grant to access a business object such as an invoice with a view entitlement.

When you generate data roles, the template applies the values of the dimension and participant data security policies to the group of base roles.

The template generates the data roles using a naming convention specified by the template's naming rule. The generated data roles are stored in the Lightweight Directory Access Protocol (LDAP) store. Once a data role is generated, you provision it to users. A user provisioned with a data role is granted permission to access the data defined by the dimension and data security grant policies of the data role template.

For example, a data role template contains an Accounts Payable Specialist role and an Accounts Payable Manager role as its base roles, and region as its dimension, with the dimension values US and UK. The naming convention is [base-role-name]:[DIMENSION-CODE-NAME]. This data role template generates four data roles.

- Accounts Payable Specialist - US (business unit)
- Accounts Payable Specialist - UK (business unit)
- Accounts Payable Manager - US (business unit)
- Accounts Payable Manager - UK (business unit)

Making Changes To Data Role Templates

If you add a base role to an existing data role template, you can generate a new set of data roles. If the naming rule is unchanged, existing data roles are overwritten.

If you remove a base role from a data role template and regenerate data roles, a resulting invalid role list gives you the option to delete or disable the data roles that would be changed by that removal.

Making Changes to Dimension Values

If you add a dimension value to your enterprise that is used by a data role template, you must regenerate roles from that data role template to create a data role for the new dimension. For example if you add a business unit to your enterprise, you must regenerate data roles from the data role templates that include business unit as a dimension.

If you add or remove a dimension value from your enterprise that is used to generate data roles, regenerating the set of data roles adds or removes the data roles for those dimension values. If your enterprise has scheduled regeneration
as an Oracle Enterprise Scheduler Services process, the changes are made automatically.

For information on working with data role templates, see the Oracle Fusion Middleware Administrator’s Guide for Authorization Policy Manager (Oracle Fusion Applications Edition).

Define Document Sequences

Document Sequences: Explained

In Oracle Fusion Applications, each business document or business event is uniquely identified by a document sequence number that you assign to it. However, the document sequencing feature must be turned on (enabled) on the business document or event to allow the assignment. For example, if document sequencing is enabled, you can assign a document sequence number to an invoice that gets generated in response to a purchase order. You can use document sequences as a proof to track successfully executed transactions as well as failed transactions. Additionally, a document sequence helps in generating an audit trail, which can be used to identify how a particular transaction passed through various applications.

Document sequencing can be managed automatically, manually, and gaplessly.

Note

Plan your document sequencing carefully before you use the options available in the application to apply sequence numbers. Avoid changes to the options after you saved your work on the Manage Document Sequences and Manage Document Sequence Categories pages.

Automatic Sequencing

Automatic document sequencing assigns a unique number to each document as it is generated, and this unique number is stored in the database. The numbering is sequential by date and time of creation. If you define a sequence to automatically number documents, you can provide an initial value to begin the sequence. In absence of a custom value, the default value 1 is used.

Manual Sequencing

Manual sequencing requires you to assign a unique number to each document before it is generated. In manual sequencing, the numerical ordering and completeness of a transaction is not enforced. Users can skip or omit numbers when entering the sequence value. However, each time that a number is assigned, the application validates its uniqueness.

Gapless Sequencing

Gapless sequencing is similar to automatic sequencing. It automatically generates a unique number for each document, but does that only for
successfully generated documents. As a result, the sequence is maintained for all the documents that are generated, and no sequence numbers are lost due to incomplete or failed document generation.

**Important**

Use this type of sequencing only if necessary because it may affect the performance of the system and slow down transaction processing.

**Sequential Numbering Enforced Profile Options**

The Sequential Numbering Enforced profile validates the documents being created and applies the relevant document sequence, based on the selected option. The following options are available:

- **Always Used:** Sequential numbering is enforced for all document categories. When this option is set, users cannot create a document if its document category is not assigned any active document sequence. The system displays an error.

- **Partially Used:** Sequential numbering is enforced for all document categories. However, if no sequence is available for the document, the system displays a warning. Users can either proceed without a document sequence or create and assign a document sequence before proceeding.

- **Not Used:** Sequential numbering is not enforced. In this mode, although the documents created do not require a document sequence to be assigned, the system checks for the uniqueness of the document number provided. The system does not display any warning or error.

**Restriction**

At the site level, the profile is by default set to Partially Used. Avoid changing this option as doing so may interfere with the validation logic and the data that was stored using the earlier option may not appear. However, you can change the option at the product or user level.

**Document Sequence Categories: Explained**

A document sequence category is a set of documents that share similar characteristics and that are formed into a logical group. Document sequence categories simplify the task of assigning number sequences to specific documents. Instead of assigning a number to each document, you assign a document sequence to one or more document sequence categories. The document sequence category automatically takes care of numbering the documents.

A document sequence category identifies the database table that stores documents resulting from transactions that your users enter. When you assign a sequence to a category, the sequence numbers the documents that are stored in a particular table. You must create document sequence categories to be able to manage the task of assigning document sequences.
Restriction

Once a document sequence category is created, you cannot change the application, the category code, or the table name. Therefore, carefully consider these details and plan your document sequencing requirement before you begin working with the application.

Once you create a document sequence category, it is available for use under the Document Sequences: Assignments section on the Manage Document Sequences page. The Category field contains the name of the document sequence category. After you create a document sequence, you can assign it to a document sequence category.

Document Sequences: Points to Consider

Sequencing documents is an important business and legal requirement. Certain aspects of the defining process are permanent and cannot be modified later. Therefore, it is important that you first decide the appropriate document sequence to use for a set of documents. You must also decide beforehand the type of document sequencing, because you are not allowed to switch to other types once a sequence is assigned to a document sequence category. Make a note of the details such as the document sequence and document sequence category so that you can refer to them at a later point in time. Also note if there are any restrictions or configuration prerequisites before you define document sequencing.

Note

Products that implement document sequencing have specifications about its usage. Refer to the corresponding product documentation for specific details and also to determine if there are any restrictions or configuration prerequisites.

Creating and Editing Document Sequences

You can create document sequences that are automatic, manual, or gapless, depending on the business or legal requirement. By default, the current date is considered as the start date. If the end date is left blank, it means that the sequence definition never expires. Among the several options used in creating and editing document sequences, the following options are functionally more important and therefore need to be carefully determined:

- **Determinant Type**: Select to limit the document sequencing activity to certain documents that belong to a specific business entity, such as Ledger, Tax Registration, and so on.

- **Initial Value**: Enter a value for the first document in your sequence. This field applies only to sequences with automatic or gapless numbering types. Sequence numbers should not be greater than eight digits. If you leave this field blank, the first document is automatically assigned a value of 1. Once a document sequence is defined, you cannot change this initial value.
Creating and Editing Document Sequence Categories

Document sequence categories are defined to make it easy to assign document sequence definitions to a group of documents instead of to individual documents. Each document sequence category is mapped to a specific table, where the documents belonging to that category are stored. The table must already be enabled for document sequencing. When specifying the table, you must consider the following points:

- When the sequential numbering feature checks for completeness or generates a report, it locates the category’s documents in the table.
- You can select only tables belonging to the application associated with the category.
- Once a category is defined, you cannot change the choice of table.

Assigning Document Sequences

Identify the documents to be numbered before assigning them a document sequence. For each document sequence, there can be only one active assignment to a document sequence category, a method code, and a determinant value (if applicable). As part of the assignment, specify whether the document is created automatically (for example, due to a batch process, or manually through a form). If you do not specify an end date, the assignment continues to remain active throughout the process cycle. If a determinant type was specified for the document sequence, then enter a specific determinant value related to the selected determinant type.

At runtime, when users create documents, the document sequence to be assigned is determined by finding the active assignment that matches the correct combination of category, numbering method, and the date range containing the transaction date.

Define Trees

Trees: Overview

Use the tree management feature in Oracle Fusion applications to organize data into hierarchies. A hierarchy contains organized data and enables the creation of groups and rollups of information that exist within an organization. Trees are hierarchical structures that enable several data management functions such as better access control, application of business rules at various levels of hierarchies, improved query performance, and so on.

For example, XYZ Corporation has two departments: Marketing and Finance. The Finance department has two functional divisions: Receivables and Payables. Defining a tree for the XYZ Corporation establishes a hierarchy between the organization and its departments, and between the departments and their respective functional divisions. Such a hierarchical modeling of organizational data could be used for executing several data management functions within that organization.
You can create one or more versions of trees, and they can be labeled for better accessibility and information retrieval. You can create trees for multiple data sources, which allow the trees to be shared across Oracle Fusion applications.

**Tree Structures**

A tree structure is a representation of the data hierarchy, and guides the creation of a tree. A tree is an instance of the hierarchy as defined in the tree structure. Tree structures enable you to enforce business rules to which the data must adhere.

The root node is the topmost node of a tree. Child nodes report to the root node. Child nodes at the same level, which report to a common parent node, are called siblings. Leaves are details branching off from a node but not extending further down the tree hierarchy.

**Tree Versions**

A tree is created having only one version. However, users can create more than one tree version depending on the need, and they can make changes to those versions. Depending on varying requirements, users can create one or more tree versions and publish all of them or some of them by making the versions active at the same time. Similar to any other version control system, versions of trees are maintained to keep track of all the changes that a tree undergoes in its life cycle.

**Tree Labels**

Tree labels are short names associated with trees and tree structures and point directly to the data source. Tree labels are automatically assigned to the tree nodes. You can store labels in any table and register the label data source with the tree structure.

**Manage Tree Structures**

**Tree Structures: Explained**

A tree structure defines the hierarchy for creating trees and prescribes rules based on which trees are created, versioned, and accessed. You can associate multiple data sources with a tree structure. A tree is an instance of this hierarchy. Every tree structure can contain one or more trees.

You can create tree structures specific to an application but you can share tree structures across applications. If you apply version control to the tree structure, it is carried over to the trees that are based on the tree structure. Each tree version contains at least one root node. Occasionally, a tree version may have more than one root node.

An administrator controls the access to tree structures through a set of rules that are periodically audited for validity.

**Tree Structure Definition: Points to Consider**

Defining a tree structure involves specifying several important pieces of information on the Create Tree Structure: Specify Definition page.
**Tree Node Selection**

The **Tree Node** table displays data in nodes that exist in the data hierarchy. You must select the correct and most appropriate tree node table to be able to define the tree structure, based on the tree hierarchy you want to establish. This selection also affects the level of security that is set on a tree node and its child entities.

**Tree Sharing Mode**

The following options are used to determine the mode of sharing a tree structure across the applications.

- **Open**: Indicates that the tree is associated with all reference data sets.
- **Set ID**: Indicates that the tree will be associated with a specific reference data set.

**Creation Mode**

Indicates the source where the tree structure is being defined. For predefined tree structures select Oracle and for custom structures, select Customers.

**Customization**

You can customize the predefined tree structures as well as the ones that you created. However, customizing the predefined tree structures involves certain level of access restrictions, and will be limited to specific tree nodes and downwards in hierarchy.

**Multiple Tree Versions**

One or more trees and tree versions can be based on a tree structure. A tree structure can have one or more trees and tree versions based on it. Usually, only one active version is permitted at any given point of time. However, depending on the requirement, you can allow two or more tree versions to be in the active state for the same date range. This flexibility allows you to choose the tree version that you want to implement.

**Managing Tree Structures: Points to Consider**

You can create, edit, and delete tree structures depending upon the requirement. You can also audit and change the status a tree structure.

**Creating and Editing Tree Structures**

You can create trees on the basis of a tree structure. When you edit an active tree structure, the status of the tree structure and all associated trees and their versions change to draft. To reuse a tree structure, you can create a copy of it without copying the associated trees and tree versions. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.
Note

For specific information on working with the predefined tree structures that exist in an Oracle Fusion application, refer to the specific product documentation.

Setting Status

If you change the status of a tree structure, the status of the trees and tree versions associated with that tree structure also changes.

The following table lists the different statuses of a tree structure.

<table>
<thead>
<tr>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>Yet to be published or is in a modified state.</td>
</tr>
<tr>
<td>Active</td>
<td>In use and based on which one or more trees or tree versions are created.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Not in use.</td>
</tr>
</tbody>
</table>

Auditing

Whenever a tree structure is set to active, the application automatically triggers an audit of that tree structure. Running an audit verifies the tree structure, checks for data integrity, and allows you to view audit details and messages and correct validation errors, if any.

Specifying Data Sources for Tree Structures: Points to Consider

The data sources provide the items for establishing hierarchy in a tree structure. In the tree management infrastructure, these data sources are Oracle Application Development Framework (ADF) business components view objects, which are defined by application development.

Labeling Schemes

Selecting a labeling scheme determines how the tree nodes are labeled. You may select a labeling scheme to assign at the data source level, at the parent node level, or keep it open for customer assignment. You may also choose not to have any labeling scheme. However, if you decide to use any of the labeling schemes, you may need to select the following additional options, to restrict the list of values that appear under the selected tree node.

- **Allow Ragged Nodes**: To include nodes that have no child nodes, and are shorter than the remaining nodes in the entire hierarchy.
- **Allow Skip Level Nodes**: To include nodes that are at the same level but have parent nodes at different levels.

Restriction of Tree Node Values

You can decide the depth of the tree structure by selecting an appropriate value from the list. Keeping the depth limit open renders an infinite list of values.
Using the following options, you can restrict the list of values that appear for selection under a specific tree node.

- **Date Range**: Specifies whether a selection of nodes should be restricted to the same date range as the tree version.
- **Allow Multiple Root Nodes**: Allows you to add multiple root nodes when creating a tree version.
- **Reference Data Set**: Specifies whether a selection of nodes should be restricted to the same set as the tree.

### Data Source Values and Parameters

Tree data sources have optional data source parameters with defined view criteria and associated bind variables. You can specify view criteria as a data source parameter when creating a tree structure, and edit the parameters when creating a tree. Multiple data sources can be associated with a tree structure and can have well-defined relationships among them.

#### Note

Parameter values customized at the tree level override the default values specified at the tree-structure level.

The data source parameters are applied to any tree version belonging to that data source, when performing node operations on the tree nodes. Data source parameters also provide an additional level of filtering for different tree structures. The tree structure definition supports three data source parameter types.

- **Bound Value**: Captures any fixed value, which is used as part of the view criteria condition.
- **Variable**: Captures and binds a dynamic value that is being used by the data source view object. This value is used by the WHERE condition of the data flow.
- **View Criteria**: Captures the view criteria name, which is applied to the data source view object.

You can also specify which of the data source parameters are mandatory while creating or editing the tree structure.

View objects from the ADF business components are used as data sources. To associate the view object with the tree structure, you can pick the code from ADF business component view objects and provide the fully qualified name of the view object, for example, oracle.apps.fnd.applcore.trees.model.view.FndLabelVO.

### Specifying Performance Options for a Tree Structure: Points to Consider

Tree structures are heavily loaded with data. As a tree management guideline, use the following settings to improve performance of data rendering and retrieval.
• Row Flattening
• Column Flattening
• Column Flattened Entity Objects
• ADF Business Component View Objects

Row Flattening

Row flattening optimizes parent-child information for run-time performance by storing additional rows in a table for instantly finding all descendants of a parent without initiating a CONNECT BY query. Row flattening eliminates recursive queries, which allows operations to perform across an entire subtree more efficiently.

To store row flattened data for the specific tree structure, users can either use the central FND_TREE_NODE_RF table or they can register their own row flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a row-flattened table contains an additional row with Corporation directly being the parent of Region (Corporation-Region).

Column Flattening

Column flattening optimizes parent-child information for run-time performance by storing an additional column in a table for all parents of a child.

To store column flattened data for the specific tree structure, users can either use the central FND_TREE_NODE_CF table or they can register their own column flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a flattened table in addition to these columns, contains three new columns: Region, Sales Division, and Corporation. Although positioned next to each other, the column Region functions at the lower level and Corporation at the higher level, retaining the data hierarchy.

Column Flattened Entity Objects

In the absence of a column-flattened table, if you need to generate the business component view objects for your tree structure for the flattened table, use the tree management infrastructure to correctly provide the fully qualified name of the entity object for the column flattened table.

ADF Business Component View Objects

View objects from the ADF business components can also be used as data sources, eliminating the need to create new types of data sources. This field is to store the fully qualified name for the business component view object generated by the tree management for business intelligence reporting and usage. The business component view object is a combination of the tree data source and column flattened entity. Using this option prevents data redundancy and promotes greater reuse of existing data, thereby improving the performance of the tree structure.
Manage Tree Labels

Tree Labels: Explained

Tree labels are tags that are stored on tree nodes. You can store labels in any table and register the label data source with the tree structure. When a labeling scheme is used for trees, the selected labels are stored in the tree label entity and each tree node contains a reference to a tree label in the labeling scheme.

The following table lists the three ways in which tree labels are assigned to the tree nodes.

<table>
<thead>
<tr>
<th>Labeling Scheme</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>Labels that are automatically assigned based on the data source to which the tree node belongs. A level label points to a specific data source. For example, in a tree that reflects the organizational hierarchy of an enterprise, all division nodes appear on one level and all department nodes on another.</td>
</tr>
<tr>
<td>Group</td>
<td>Labels that you can arbitrarily assign to tree nodes.</td>
</tr>
<tr>
<td>Depth</td>
<td>Labels that are automatically assigned based on the depth of the tree node within the tree. No manual assignment is performed.</td>
</tr>
</tbody>
</table>

Note

In an unbalanced hierarchy, a level may not be equal to depth.

Manage Trees and Tree Versions

Managing Trees and Tree Versions: Points to Consider

You can create and edit trees and tree versions depending upon the requirement. A tree can have one or more tree versions. Typically, when changes are made to an existing tree, a new version is created and published.

Creating and Editing Trees

Trees are created based on the structure defined in the tree structure. You can create trees, modify existing trees, and delete trees. If you want to copy an existing tree, you can duplicate it. However, only the tree is duplicated and not its versions.

Creating a tree involves specifying the tree definition and specifying the labels that are used on its nodes. If the selected tree structure has data sources and parameters defined for it, they appear on the page allowing you to edit the parameter values at the tree node level.

Note
Parameter values customized at the tree level will override the default values specified at the tree-structure level.

**Creating and Editing Tree Versions**

Tree versions are created at the time of creating trees. A tree must contain a version.

Editing an existing tree provides you the choice to update the existing version. You can also edit the existing version that lies nested under the tree in the search results.

When you edit a tree version bearing Active status, the status changes to Draft until the modifications are saved or cancelled.

**Trees and Data Sources: How They Work Together**

Data sources form the foundation for tree management in Oracle Fusion Applications. Tree structures, trees, and tree versions establish direct and real-time connectivity with the data sources. Changes to the data sources immediately reflect on the Manage Trees and Tree Versions page and wherever the trees are being used.

**Metadata**

Tree structures contain the metadata of the actual data that is used in Oracle Fusion Applications. Tree structures contain the core business logic that is manifested in trees and tree versions.

**Data Storage**

Trees and tree versions are built upon the tree structures. They employ the business rules defined in the tree structures and allow an application to select and enable a subset of trees to fulfill a specific purpose in that application.

**Access Control**

Source data is mapped to tree nodes at different levels in the database. Therefore, changes you make to the tree nodes affect the source data. Access control set on trees prevents unwanted data modifications in the database. Access control can be applied to the tree nodes or anywhere in the tree hierarchy.

**Adding Tree Nodes: Points to Consider**

Tree nodes are points of data convergence that serve as the building blocks of a tree structure. Technically, the node may be stored either in a product-specific table or in an entity that has been established by tree management as the default storage mechanism. However, since all data in Oracle Fusion Applications usually have a storage home, only user-created data needs to be stored in an entity.

Nodes are attached to tree versions. Whenever you create or edit a tree version, you need to specify its tree node.
Managing Tree Nodes

You can create, modify, or delete tree nodes on the Tree Version: Specify Nodes page. To add a tree node, ensure that the tree structure with which the tree version is associated is mapped to a valid data source. You can also duplicate a tree node if the multiple root node feature is enabled.

Node Levels

In most trees, all nodes at the same level represent the same kind of information. For example, in a tree that reflects the organizational hierarchy, all division nodes appear on one level and all department nodes on another. Similarly, in a tree that organizes a user’s product catalog, the nodes representing individual products might appear on one level and the nodes representing product lines on the next higher level.

When levels are not used, the nodes in the tree have no real hierarchy or reporting structure but do form a logical summarization structure. Strictly enforced levels mean that the named levels describe each node’s position in the tree. This is natural for most hierarchies. Loosely enforced levels mean that the nodes at the same visual level of indentation do not all represent the same kind of information, or nodes representing the same kind of information appear at multiple levels. With loosely enforced levels, users assign a level to each node individually. The level is not tied to a particular visual position.

Node Types

A tree node has the following node types.

- Single: Indicates that the node is a value by itself.

- Range: Indicates that the node represents a range of values and possibly could have many children. For example, a tree node representing account numbers 10000 to 99999.

- Referenced Tree: Indicates that the tree node is actually another version for the tree based on the same tree structure, which is not physically stored in the same tree. For example, a geographic hierarchy for the United States can be referenced in a World geographic hierarchy.

Define Profile Options

Profile Options: Explained

Profile options manage configuration data centrally and influence the behavior of applications. Profile options serve as permanent user preferences and application configuration parameters. You configure profile options with settings for specific contexts or groups of users. Users customize how their user interfaces look and behave by changing the values of available profile options.

Profile options store the following kinds of information.
You can add and configure new profile options in addition to configuring predefined profile options that are implemented as updateable.

### Profile Option Definition and Configuration

Application developers add new profile options and configure ones that are not to be updated by other users. Application administrators and implementation consultants configure profile options with profile option values that are implemented as updatable.

Profile option definitions consist of the following.

- Profile option name
- Application and module in the application taxonomy
- Profile option values
- Profile options categories
- Profile option levels
- Profile option level hierarchy

Profile options can appear on any user interface page without indication that a profile option is what is being set.

### Profile Option Values

Some profile options have predefined profile option values.

The Manage Profile Option Values task flow allows an administrator to set updatable profile option values at the available levels, including the user level. You can access the Manage Profile Option Values task starting in the Setup and Maintenance Overview page and searching for profile option tasks.

You can set profile option values at different levels: site, product, and user. The following table provides examples.

<table>
<thead>
<tr>
<th>Profile Option Level</th>
<th>Value of the Profile Option Level</th>
<th>Profile Option Value</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Manager1</td>
<td>UK pound sterling</td>
<td>Access to site and all products shows UK pounds sterling in effect</td>
</tr>
</tbody>
</table>
Context such as user session or accessed product determines which profile option value is associated with the profile option name. In the example, if manager1 does not set a profile option value for this profile option, access to Financials for EMEA shows currency in Euros; and access to other products shows currency in UK pounds sterling.

**Profile Option Categories**

Categories group profile options based on their functional area. Profile option categories facilitate searching and defining data security.

For example, in Oracle Fusion Receivables, the Transactions profile option category groups profile options related to setting how Receivables transactions are to be processed, such as Require Adjustment Reason.

A profile option can be in more than one category.

**Profile Option Hierarchies and Levels**

Application developers specify at which hierarchy level a profile option is enabled. The predefined profile option hierarchy levels are site, product, and user.

The hierarchy levels specified in the profile option definition determine the context in which a profile option value may be set. If the profile option value at a particular level is updatable, an administrator can update the profile option value for that context.

**Note**

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

For security, one level in the hierarchy is designated as a user level. A profile option may be enabled at any or all hierarchy levels. When enabled at all levels, the predefined ordering of profile option hierarchy levels gives precedence to the values that are set at the user level over values set at the product and site levels, and precedence to values set at the product level to values set at the site level. If there is no value for the current user, then the product value applies. If there is no value for the user or product, then the site value applies.
The table shows the predefined profile option hierarchy and ordering.

<table>
<thead>
<tr>
<th>Hierarchy Level</th>
<th>Priority When Multiple Levels Set</th>
<th>Effect on Applications</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>Lowest</td>
<td>Affect all applications for a given implementation</td>
<td>Currency for the site is set to Euros.</td>
</tr>
<tr>
<td>Product</td>
<td>Supersedes Site</td>
<td>Affect all applications of a product family such as Financials</td>
<td>Currency for the Financials products set to UK pound sterling.</td>
</tr>
<tr>
<td>User</td>
<td>Highest, supersedes Product</td>
<td>Affect only the experience of the current user</td>
<td>Currency for the user of Financials applications set to US dollars.</td>
</tr>
</tbody>
</table>

You can configure updatable values for profile options at one or more levels depending on which levels are enabled in the profile option definition. When a profile is set at more than one level, higher levels of specificity override lower levels of specificity.

In the example, if the currency setting for the site is UK pounds sterling, but the Financials division works in the Netherlands using the euro, a manager in the US can override that product level setting at the user level to use US dollars when accessing Financials applications.

In another example, if a profile option called Printer is set only at the site and product levels. When a user logs on, the Printer profile option assumes the value set at the product level, since it is the highest level setting for the profile.

Tip

Set site-level profile option values before specifying values at any other level. The profile option values specified at the site-level work as defaults until profile option values are specified at the other levels.

**Planning Profile Options: Points to Consider**

Plan profile options before defining and configuring them.

The following aspects assist you in better planning how to manage profile options.

- Profile option tasks
- Before creating a profile option
- Profile options data model

**Profile Option Tasks**

Users may be able to set their own profile options, depending on settings in the profile option definition. However, not all profile options are visible to end users, and some profile options, while visible, may not be updated by end users.

The following table lists tasks and considerations relevant to planning profile options.
Define Common Applications Configuration for Marketing

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Role</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning, creating, and editing a new profile option</td>
<td>Applications developer</td>
<td>Since profile options are for permanent settings, do not use profiles options to cache temporary session attributes. Add capacity for user preferences and system configuration. Customize profile options with values, value behaviors, validation, category values, and security. Define the levels at which the profile option is enabled.</td>
</tr>
<tr>
<td>Configure values in an existing profile option</td>
<td>Applications developer, application administrator, and implementation consultant</td>
<td>Manage the values for existing profile options.</td>
</tr>
<tr>
<td>Create and edit profile option categories</td>
<td>Applications developer, application administrator, and implementation consultant</td>
<td>Manage categories for organizing existing profile options.</td>
</tr>
</tbody>
</table>

**Note**

Since a profile option enables a behavior in an application user interface or across applications, a value change made by an end user is reflected in the UI page for managing profile option values.

**Before Creating a Profile Option**

Profile options are best defined for managing configuration data centrally and influencing the behavior of applications.

If the purpose of a profile option setting is specific to a piece of data (typically setup data), it is best implemented as an attribute of that data.

Do not use profile options for behavior that is not configurable.

Profile options exist independent of role.

Do not use profile options to implement function security. For example, an application should not check for a profile option value set to yes to provide access to a page. Do not use profile options to implement data security, such as a profile option value that must be set to a specific value to provide view access to an entity.

Do not use profile options to capture a dynamic system states, such as data stored in a temporary table. Use Global Variables for temporary states instead.

Evaluate if there is a genuine need before creating a profile option. Do not force users to make a decision about an aspect of their application use that is of no concern.

Evaluating need includes looking for duplicate or similar profile options, even in other products, before creating a new one. For example, you do not need multiple profile options to choose a preferred currency.
Profile Options Data Model

The profile option data model illustrates the relationships among profile option elements.

The figure shows the data model of profile option entities.

For more information about planning profile options, see the Oracle Fusion Applications Developer’s Guide.

Managing Profile Options: Points to Consider

A profile option definition consists of a name for the profile option and valid values. It is defined within a module of the application taxonomy. Application developers manage profile options to create new profile options or modify existing profile option definitions, which includes specifying the levels at which a profile option is enabled and defining values. Implementation consultants and application administrators configure existing profile options by managing the profile option’s updatable values, and creating categories that group profile options.

Configuring a Profile Option

A profile option definition includes information about the owning application and module in the application taxonomy. A start or end date, or both may limit
when a profile option is active. The profile option definition may include an SQL validation statement that determines which values are valid, and the hierarchy levels at which the profile option is enabled and updatable.

To be visible to users, a profile option must be user enabled. You can also allow user updates of the profile option, which means users can make changes to the validation and the profile option level information.

Profile option levels specify at which context level profile values may be enabled or updated.

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

**SQL Validation**

The SQL validation of the profile option definition determines what valid profile option values are available. In the absence of validation, any value is valid.

For example, SQL validation provides a means of defining a list of values for the valid values of the profile option. The SQL validation can use lookups to provide the valid values for profile options, such as the lookup codes of the YES_NO lookup type.

With a profile option called DEFAULT_LANGUAGE, you can configure the following validation.

```
SELECT DESCRIPTION Language, NLS_LANGUAGE
FROM FND_LANGUAGES_VL
WHERE INSTALLED_FLAG IN ('B','I')
ORDER BY DESCRIPTION
```

This results in the following list of values based on data in `FND_LANGUAGE_VL`.

<table>
<thead>
<tr>
<th>Display Value</th>
<th>Hidden Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>American English</td>
<td>US</td>
</tr>
<tr>
<td>French</td>
<td>F</td>
</tr>
<tr>
<td>Spanish</td>
<td>E</td>
</tr>
</tbody>
</table>

Hidden values must be varchar2(2000).

Profile options generally provide configuration values within a particular context. Though you can create a profile option to be global, think of global values as default values to avoid storing inappropriate configuration information as profile option values. Create global profile options that have corresponding contextual levels.

**Managing Profile Option Categories: Points to Consider**

Use profile option categories to group profile options.
Organizing Profile Options in Categories

As a guideline, group profile options in a single category if the profile options affect the same feature, or if an administrator would likely want to see the profile options in the results of a single search.

Application developers are responsible for the initial groupings and then administrators can make changes based on their specific needs. Administrators can categorize profile options and then easily search on profile options by category.

Tip

Define profile option categories first and assign new profile options to existing categories rather than defining profile options first and then defining categories to categorize them.

Adding New Profile Option Categories

You can add new categories or add profiles to an existing category.

You can create a profile option category by duplicating an existing category and editing it for a new grouping of profile options. You can add multiple profile options to a category. A profile option can exist in multiple categories.

Profile Option Order in a Category

Specify a profile option sequence to determine the order of profile options when queried by profile option category.

Viewing and Editing Profile Option Values: Points to Consider

A profile option value consists of the value and the context or level where the value is set. You specify the context with a pairing of the profile option value’s level and level value, such as the product level and the level value GL for Oracle Fusion General Ledger. Adding or modifying profile option values can include deciding which valid values are enabled or updatable at which level.

The SQL validation of the profile option definition determines what valid profile option values are available. In the absence of validation, any value is valid.

Profile Option Levels and User Session Context

Site level profile option values affect the way all applications run for a given implementation. Product level profile option values affect the way applications owned by a particular product code behave. For example, a product may use profile options set at the product level to determine how regions provided by a common module such as those available from Oracle Fusion Trading Community Model or CRM display in a particular work area or dashboard. User level profile option values affect the way applications run for a specific application user.
Whichever profile option value is most specific to a user session, that is the value at which the profile option is set for the user session.

For example, the predefined FND_LANGUAGE profile option sets the default language. In addition to a site level value, you can define a value for various product or user levels.

<table>
<thead>
<tr>
<th>Level Name</th>
<th>Level Value</th>
<th>Profile Option Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>InFusion</td>
<td>American English</td>
</tr>
<tr>
<td>Product</td>
<td>Customer Center</td>
<td>French</td>
</tr>
<tr>
<td>Product</td>
<td>CRM Application Composer</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Application Administrator</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Hima</td>
<td>Hindi</td>
</tr>
</tbody>
</table>

Values at the site level take affect for any user unless overridden by a different value set at the more specific levels of product and user. Product level profile option values affect the way applications owned by a particular product code behave. In addition to user level profile option values in applications, selections may be available in the user preferences workspace.

The profile option values for various user session contexts are as follows.

<table>
<thead>
<tr>
<th>Site</th>
<th>Product</th>
<th>User</th>
<th>Highest Available Level</th>
<th>Active Profile Option Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>InFusion</td>
<td>CRM Application Composer</td>
<td>Hima</td>
<td>User</td>
<td>Hindi</td>
</tr>
<tr>
<td>Acme</td>
<td>Payables</td>
<td>Application Administrator</td>
<td>User</td>
<td>American English</td>
</tr>
<tr>
<td>InFusion</td>
<td>Customer Center</td>
<td>Guillaume</td>
<td>Product</td>
<td>French</td>
</tr>
<tr>
<td>InFusion</td>
<td>Payables</td>
<td>Implementation Consultant</td>
<td>Site</td>
<td>American English</td>
</tr>
<tr>
<td>Acme</td>
<td>Payables</td>
<td>Implementation Consultant</td>
<td>none</td>
<td>no value</td>
</tr>
</tbody>
</table>

**Note**

More than one site level value is relevant in an enterprise with multiple tenants using a single instance of Oracle Fusion Applications.

**Effect of Changes to Profile Option Values**

Any change you make to a user level profile option has an immediate effect on the way applications run for that session. When you sign in again, changes made to your user level profile options in a previous session are still in effect. When you change profile option value at the product level and no user level values are set, you see the update immediately, but other users may not see the changed value until signing out and back in. When you change a profile option value and the new value affects other users, the change takes effect only when users sign in the next time.
Changes to site level profile options take effect for any user session that is started after the setting has been changed. Changes to site or user level profile options do not affect any user sessions that are already in progress when the change is made.

Changes to site or user level profile options take effect for any C or PL/SQL processes, such as scheduled jobs, that are launched after the setting has been changed. Profile option changes do not affect C or PL/SQL processes that are already running.

Define Flexfields

Flexfields: Overview

A flexfield is an extensible data field that is divided into segments and available for capturing enterprise specific information. Each segment captures a single atomic value, which is represented in your application database as a single column.

Flexfields allow enterprise requirements to be met without changing the data model. Different data can be captured on the same database table.

The following three types of flexfields are available in Oracle Fusion Applications and provide a means to customize applications features without programming.

- Key
- Descriptive
- Extensible

Flexfields encapsulate all of the pieces of information related to a specific purpose, such as a key identifying a particular purchase, or the components of a student’s contact information, or the features of a product in inventory. Setup of key flexfields is generally required for correct operations of a product. In the case of descriptive and extensible flexfields, segments store attributes on an entity to capture additional information, so setup is usually optional.

End users see flexfield segments as attributes of information displayed in the application user interface. They enter a value for the attribute. End users do not modify the configuration of attributes; they enter values only for attributes that are already configured.

Segments

All flexfields consist of segments. Segments represent attributes of information. They can appear globally wherever the flexfield is implemented, or based on a structure or context.

Value Sets

A value set is a predefined, named group of values that can be used to validate the content of a flexfield segment. The value set you assign to a flexfield segment defines the valid values for the attribute represented by that segment.
Structure

Key flexfields have structure, or a specific configuration of segments. Adding or removing segments, or rearranging their order, produces a different structure. A key flexfield can have multiple structures if registered to support more than one structure. Each instance of a structure shares the same number and order of segments, but differs in the allowable values or value sets that validate the segments.

In some applications, different users need to see different segment structures for the same flexfield. For example, the correctly formatted local postal address for customer service inquiries differs based on locale. The postal address key flexfield can display different segments and prompts for different end users based on a data condition in your application data, such as the user’s role or a value entered by the user.

Context

Descriptive flexfield segments can be context-sensitive and extensible flexfield segments must be context-sensitive. Segments are made available to an application as groups of attributes called contexts. A context is a set of context-sensitive segments that store a particular type of related information.

You define contexts as part of configuring a descriptive or extensible flexfield.

The database columns on which context-sensitive segments are based can be reused in as many contexts as desired.

The same column can be used by a similar segment in different contexts. For example, you can define a Dimensions context that consists of segments representing height, width and depth. You can also define a Measurements context that contains segments which reuse the same underlying height, width and depth columns for length values on an x, y, and z axis, and which additionally includes segments for weight, volume and density.

The same underlying column can be used by different segments in different contexts. For example, you can define a Dimensions context that uses the ATTRIBUTE1 column for height, the ATTRIBUTE2 column for width, and the ATTRIBUTE3 column for depth. You can also define a Measurements context that uses the same columns for other attributes: the ATTRIBUTE1 column for weight, the ATTRIBUTE2 column for volume, and the ATTRIBUTE3 column for density.

Usage

Usage affects various aspects of flexfields. The usage of the flexfield is set when the flexfield is registered and specifies the application and table with which the flexfield is associated.

Entity usage indicates the table containing the segments of a flexfield.

A flexfield can have multiple usages. The first table registered for a flexfield is the master usage. Segments are based on the master usage, and other usages of the same table for the same flexfield use the same segment setup, though the column names optionally may have a differentiating prefix.
You can configure different behavior for extensible flexfield contexts at the usage level. The usage of an extensible flexfield context determines in which scenarios or user interfaces the segments of a context appear to end users. For example, if a Supplier page displays an extensible flexfield’s supplier usage and a buyer page displays that same extensible flexfield’s buyer usage, a context that is associated to the supplier usage but not the buyer usage displays only on the supplier page and not the buyer page.

The usage of value sets specifies the segments where the value set is assigned.

**Deployment**

A flexfield must be deployed to display its current definition in a runtime application user interface. For example, if the deployment status is Edited, the flexfield segments may already appear in the UI based on the flexfield definition at the time of last deployment.

You can deploy a flexfield as a sandbox for testing the configuration before deploying it to the mainline for test or production users.

**Runtime Appearance**

In an application user interface, flexfield segments appear as label and field pairs or as a table of fields where the column headers correspond to the labels. The fields represent the flexfield segments and accept entered input or a selection from a list of choices that correspond to the segment's assigned value set. Extensible flexfield segments appear grouped within labeled regions, where each grouping is a context and the region labels are the context names.

All segments of a single flexfield are grouped together by default. The layout and positions of the flexfield segments depend on where the application developer places the flexfield on the page. Flexfields may also be presented in a separate section of the page, alone in a table, or on their own page.

You can use Oracle Composer to edit the layout, position, or other display features of the flexfield segments.

Manage flexfields using the tasks of the Define Flexfields activity, which you can access by starting in the Setup and Maintenance Overview page and searching for flexfield tasks.

**Flexfields and Oracle Fusion Application Architecture: How They Work Together**

Administrators configure flexfield segments to capture data that represents the values of attributes. Flexfield segments represent attributes of entities (business objects). Most business objects are enabled for descriptive flexfields. Some business objects are enabled for extensible flexfields.

The figure shows the layers of a flexfield: the business entity table and metadata in the database, business components that are Application Development Framework (ADF) objects or ADFbc objects derived from the metadata and stored in the Metadata Services Repository (MDS), and the user interface where the input fields defined by the flexfield segments are rendered. The flexfield definition consists of all the metadata defined during configuration and stored in the database.
Application developers create a flexfield and register it so that it is available for configuration. Administrators and implementation consultants configure segments and other properties of the available flexfields. This information is stored as additional flexfield metadata in the database. Deploying the flexfield generates ADF business components based on the flexfield metadata in the database.

**Accessing Flexfields For Configuration**

A flexfield affects the Web Services Description Language (WSDL) schemas exposed by ADF services and used by SOA composites. The Web services that expose base entity data also expose flexfield segment data.

Attributes incorporate into SOA infrastructure (BPEL, Rules) and integrate with Oracle Business Intelligence Enterprise Edition and Extended Spread Sheet Database (ESSbase). This ensures that flexfields are preserved across Oracle Fusion Applications updates.
Deployment

The metadata for the flexfield is stored in the application database as soon as you save your configuration changes. Deploying the flexfield generates the ADF business components so that the runtime user interface reflects the latest definition of the flexfield in the metadata.

Runtime

For a flexfield to reflect the latest flexfield definition at runtime it must be deployed. The user interface accesses a business object and the deployed flexfield definition indicates which business object attributes the flexfield captures values for. If you add display customizations for a flexfield using Oracle Composer, these are customizations on the page so that the same flexfield segments can appear differently on various different pages.

Values entered for segments are validated using value sets.

Managing Flexfields: Points To Consider

Managing flexfields involves, registering, planning, and configuring flexfields.

You plan and configure the registered flexfields provided in your applications by applications developers. How you configure flexfield segments determines how the flexfield segments appear to end users. Optionally, you can customize the UI page to change how the flexfield segments appear to end users on that page.

The figure shows the processes involved in making flexfields available to end users. The tasks in the Define Flexfields activity allow administrators to configure and deploy flexfields. If you deploy a flexfield to a sandbox and decide not to make further changes, you select the flexfield in the Manage Flexfields tasks of the Define Flexfields activity and deploy the flexfield in the mainline so it is available to users.

**Registering Flexfields**

Application development registers flexfields so they are available to administrators and implementation consultants for configuration.

Application development creates the capacity of database tables to support flexfields so an enterprise can capture specific information about an entity. Many flexfields are predefined in Oracle Fusion Applications.

A flexfield must be registered before it can be configured.

For more information on registering flexfields, see Oracle Fusion Applications: Developer’s Guide.

**Planning Flexfields**

All flexfields consist of segments which represent attributes of an entity. The values an end user inputs for an attribute are stored in a column of the entity.
table. Application development enables columns of entity tables for use in flexfields during flexfield registration.

A flexfield must be registered before it can be configured. Before configuring new flexfield segments for your enterprise, be sure to plan their implementation carefully.

For more information on planning flexfield configuration, see Oracle Fusion Applications Extensibility Guide.

Configuring Flexfields

Administrators or implementers configure flexfields so they meet the needs of the enterprise. Some flexfields require configuration to make an application operate correctly.

Configuring a flexfield includes the following:

- Defining value sets against which the values entered by end users are validated
- Defining the structure or context of the segments in the flexfield
- Specifying the prompt, length and data type of each flexfield segment
- Specifying valid values for each segment, and the meaning of each value within the application

Tip

As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

Some descriptive and extensible flexfields provide parameters, which are attributes of the same or related entity objects. You use parameters to set the initial value or derivation value of an attribute from external reference data, such as a column value or a session variable, rather than from user input.

Note

Adding segments to represent additional attributes is considered a custom task. For more information, see the Oracle Fusion Applications Extensibility Guide.

For more information on adding columns to a table, see the Oracle Fusion Applications Developer’s Guide.

For more information on configuring flexfields for custom attributes, see also the Oracle Fusion Applications Extensibility Guide.

Enabling a Flexfield for Business Intelligence

You can enable key flexfields segment instances and descriptive flexfield segments for business intelligence if the flexfield is registered in the database as an Oracle Business Intelligence-enabled flexfield. For more information on enabling segments for business intelligence, see points to consider for managing key and descriptive flexfields.

If you BI-enable multiple segment instances from the same flexfield, the business components of the flexfield are flattened automatically to include a separate attribute for each of the BI enabled segment instances, even if some of them serve the same purpose in your application. You can prevent this duplication and the
extra workload and complexity that result, by enabling equalization for any set of segment instances that serve the same purpose in different structure instances. Segment instances that are equalized will appear in the flattened flexfield as a single attribute.

**Deploying Flexfields**

Once you have configured a flexfield, you must deploy it to make the latest definition available to end users.

You deploy a flexfield in the mainline for general use in a test or production environment, or you can deploy a flexfield as a flexfield-enabled sandbox to confirm that it is correctly configured before deploying it to the mainline.

Deploying a flexfield results in a deployment status. Once deployed, the deployment status indicates the state of the currently configured flexfield relative to the last deployed definition.

**Optionally Changing How Flexfield Segments Appear in a User Interface Page**

The flexfield attributes you define integrate with the user interface pages where users access the attributes' business object. Application development determines the UI pages where business objects appear and the display patterns used by default to render flexfield segments.

You can customize the appearance of the flexfield segments in the UI page using Oracle Composer once the flexfield is deployed to the mainline.

For more information on customizing flexfield appearance with Oracle Composer, see guidance on customizing existing pages in Oracle Fusion Applications Extensibility Guide.

For more information on planning flexfields or customizing them beyond the configuration possible in the Define Flexfields tasks, see Oracle Fusion Applications Extensibility Guide.

For more information on creating flexfields and adding them to a UI page, see Oracle Fusion Applications Developer's Guide.

**Flexfield Segment Properties: Explained**

Independent of the value set assigned to a segment, segments may have properties such as validation.

**Range Validation of Segments**

Range validation enables you to enforce an arithmetic inequality between two segments of a flexfield. For example, a product must be ordered before it can be shipped, so the order date must be on or before the ship date, and consequently the order date segment value must be less than or equal to the ship date segment value. You can use range validation to ensure this relationship.

The conditions for range validation are as follows.

- Segments must be configured for range validation in pairs, one with the low value and one with the high value.
- Both segments must be the same data type.
- Both segments must be part of the same structure in a key flexfield or part of the same context in a descriptive flexfield or extensible flexfield.
- The low value segment must have a lower sequence number than the high value segment.

- Non-range validated segments can exist between a range validated pair, but range validated pairs cannot overlap or be nested.

You can configure as many range validated pairs as you want within the same flexfield. Your application automatically detects and applies range validation to the segment pairs that you define, in sequence order. It must encounter a low value segment first, and the next range validated segment it encounters must be a high value segment. These two segments are assumed to be a matching pair. The low value and the high value can be equal.

**Flexfields and Value Sets: How They Work Together**

Value sets are specific to your enterprise. When gathering information using flexfields, your enterprise's value sets validate the values your users enter based on how you defined the value set.

You can assign a value set to any number of flexfield segments in the same or different flexfields. Value set usage information indicates which flexfields use the value set.

**Defining and Assigning Value Sets**

As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

**Caution**

Be sure changes to a shared value set are compatible with all flexfield segments using the value set.

**Shared Value Sets**

When you change a value in a shared value set, the change affects the value set for all flexfields that use that value set. The advantage of a shared values set is that a single change propagates to all usages. The drawback is that the change shared across usages may not be appropriate in every case.

**Deployment**

When you deploy a flexfield, the value sets assigned to the segments of the flexfield provide end users with the valid values for the attributes represented by the segments.

**Defaulting and Deriving Segment Values: Explained**

To populate a flexfield segment with a default value when a row is created, specify a default type of constant or parameter and a default value.

To synchronize a segment’s value with another field’s value whenever it changes, specify the derivation value to be the flexfield parameter from which to derive the attribute’s value. Whenever the parameter value changes, the attribute's value is changed to match. If you derive an attribute from a parameter, consider making the attribute read-only, as values entered by users are lost whenever the parameter value changes.

When defaulting or deriving a default value from a parameter, only those attributes designated by development as parameters are available to be chosen.
Different combinations of making the segments read only or editable in combination with the default or derivation value or both, have different effects.

Initial runtime behavior corresponds to the row for the attribute value being created in the entity table. If the default value is read only, it cannot subsequently be changed through the user interface. If the default value is not read only, users can modify it. However, if the segment value is a derived value, a user-modified segment value is overwritten when the derivation value changes.

<table>
<thead>
<tr>
<th>Default Type</th>
<th>Default value specified?</th>
<th>Derivation value specified?</th>
<th>Initial runtime behavior</th>
<th>Runtime behavior after parameter changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No</td>
<td>Yes</td>
<td>No initial segment value</td>
<td>Changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Constant</td>
<td>Yes</td>
<td>No</td>
<td>Default segment value</td>
<td>N/A</td>
</tr>
<tr>
<td>Constant</td>
<td>Yes</td>
<td>Yes</td>
<td>Default segment value</td>
<td>Changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>No</td>
<td>Default segment value is the parameter's default value</td>
<td>N/A</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and same as default value</td>
<td>Default segment value is parameter's default and derivation value</td>
<td>Changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and different from default value</td>
<td>Default segment value is parameter's default value</td>
<td>Changed parameter derivation value does not update segment value. Only the changed derivation value updates the segment value.</td>
</tr>
</tbody>
</table>

**Define Flexfields: Flexfield Deployment**

**Flexfield Deployment: Explained**

To use a flexfield at runtime, the flexfield must have been deployed at least once. Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. Flexfields are deployed for the first time during the application provisioning process.

You can deploy a flexfield to a sandbox for testing or to the mainline for use.

**Deployment Status**

Every flexfield has a: deployment status.

A flexfield can have the following deployment statuses.
<table>
<thead>
<tr>
<th>Deployment Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edited</td>
<td>The flexfield metadata definition has not been deployed yet. Updates of the metadata definition are not applied in the runtime yet.</td>
</tr>
<tr>
<td>Patched</td>
<td>The flexfield metadata definition has been modified through a patch, but the flexfield has not yet been deployed so the patched definition is not reflected in the runtime.</td>
</tr>
<tr>
<td>Deployed to Sandbox</td>
<td>The current metadata for the flexfield is deployed in ADF artifacts and available as a flexfield-enabled sandbox. The status of the sandbox is managed by the Manage Sandboxes task available to the Administrator menu of Setup and Maintenance work area.</td>
</tr>
<tr>
<td>Deployed</td>
<td>The current metadata for the flexfield is deployed in ADF artifacts and available to end users. There have not been any changes to the flexfield since it was last deployed in the mainline.</td>
</tr>
<tr>
<td>Error</td>
<td>The deployment attempt in the mainline failed.</td>
</tr>
</tbody>
</table>

**Note**
Whenever a value set definition changes, the deployment status of a flexfield that uses that value set changes to edited. If the change results from a patch, the deployment status of the flexfield changes to patched.

**Initial Deployment Status of Flexfields**

The Oracle Fusion Applications installation loads flexfield metadata into the database. This initial load sets the flexfield status to Edited to indicate that the flexfield has not been deployed yet. The application provisioning process during installation deploys the predefined flexfields of the provisioned applications, which sets their status to Deployed if no errors are encountered.

When accessing a provisioned application, deployed flexfields are ready to use. In some cases, flexfield availability at runtime requires setup, such as defining key flexfields.

**Metadata Validation**

Use the Validate Metadata command to view possible metadata errors before attempting to deploy the flexfield. Metadata validation is the initial phase of the Deploy and Deploy to Sandbox commands. By successfully validating metadata before running the deployment commands, you can avoid failures in the metadata validation phase of a deployment attempt. Errors in the metadata validation phase of deployment cause the deployment attempt to abort. Metadata validation results do not affect the deployment status of a flexfield.

**Flexfield Deployment Status: How It Is Calculated**

Flexfield deployment status indicates how the flexfield metadata definition in the Oracle Applications database relates to the Application Development
Framework (ADF) business components generated into a Metadata Services (MDS) repository.

**Settings That Affect Flexfield Deployment Status**

If you have made a change to a flexfield and expect a changed deployment status, be sure you have saved your changes. No settings affect flexfield deployment status.

**How Flexfield Deployment Status Is Calculated**

If the flexfield definition has been edited through the Define Flexfields activity task flows, the status is Edited. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. Any change, including if a value set used in a flexfield changes, changes the deployment status to Edited. If a flexfield has never been deployed, its status is Edited.

*Note*

When an application is provisioned, the provisioning framework attempts to deploy all flexfields in that application.

If you deploy the flexfield to a sandbox successfully, the status is Deployed to Sandbox. The latest flexfield metadata definition in the Oracle Fusion Application matches the metadata definition that generated ADF business components in a sandbox MDS repository. Whether the sandbox is active or not does not affect the deployment status. If the flexfield was deployed by a sandbox and has not been edited or re-deployed to the mainline since then, the status remains Deployed to Sandbox independent of whether the sandbox is active, or who is viewing the status.

If you deploy the flexfield successfully, meaning to the mainline, the status is Deployed. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a mainline MDS repository. Change notifications are sent when a flexfield is deployed successfully to the mainline.

If either type of deployment fails so the current flexfield definition is not deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest successfully deployed flexfield definition.

If the flexfield definition has been modified by a patch, the status is Patched. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition.

When a deployment attempt fails and you can access the Deployment Error Message for details.

**Deploying a Flexfield-Enabled Sandbox: How It Works With Mainline Metadata**

The flexfield definition in a sandbox corresponds to the flexfield metadata definition in the Oracle Fusion Applications database at the time the flexfield was deployed to the sandbox. When the flexfield is ready for end users, the flexfield must be deployed to the mainline.

A flexfield-enabled sandbox uses the following components.
• Flexfield metadata in the Oracle Fusion Applications database
• Flexfield business components in a sandbox Metadata Services (MDS) repository
• User interface customizations for the flexfield in the mainline MDS repository

The figure shows the two types of deployment available in the Manage Flexfield tasks of the Define Flexfields activity. Deploying a flexfield to a sandbox creates a sandbox MDS repository for the sole purpose of testing flexfield behavior. The sandbox is only accessible to the administrator who activates and accesses it, not to users generally. Deploying a flexfield to the mainline applies the flexfield definition to the mainline MDS repository where it is available to end users. After deploying the flexfield to the mainline, customize the page where the flexfield segments appear. Customization of the page in the sandbox MDS repository cannot be published to the mainline MDS repository.
Sandbox Metadata Services Repository Data

Deploying the flexfield to a sandbox generates the Application Development Framework (ADF) business components of a flexfield in a sandbox MDS repository for testing in isolation.

Warning

Do not make changes to flexfield segment display features in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline.

Mainline Metadata Services Repository Data

The Oracle Fusion Applications database stores the single source of truth about a flexfield. From this the ADF business component objects that implement the flexfield in the runtime user interface are generated in the mainline MDS repository when the flexfield is deployed.

Deploying a Flexfield-Enabled Sandbox: Points to Consider

Deploying a flexfield to a sandbox creates a flexfield-enabled sandbox. Each flexfield-enabled sandbox contains only one flexfield.

You can test the runtime behavior of a flexfield in the flexfield-enabled sandbox. If changes are needed, return to the Define Flexfield tasks to change the flexfield definition.

When you deploy a flexfield to sandbox, the process reads the metadata about the segments from the database, generates flexfield Application Development Framework (ADF) business component artifacts based on that definition, and stores in the sandbox only the generated artifacts derived from the definition.

Sandbox MDS Repository Data

The sandbox data allows you to test the flexfield in isolation without first deploying it in the mainline where it could be accessed by users.

Warning

Do not make changes to flexfield segment display features in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline.

Managing a Flexfield-Enabled Sandbox

When you deploy a flexfield as a sandbox, that flexfield-enabled sandbox automatically gets activated in your user session. When you sign back in to see the changes, the sandbox is active in your session.

You can only deploy a flexfield to a sandbox using the Define Flexfields task flow pages.
You also can use the Manage Sandboxes feature in the Administration menu of the Setup and Maintenance work area to activate, access, or delete a flexfield-enabled sandbox.

**Note**

Whether you use the Define Flexfields or Manage Sandboxes task flows to access a flexfield-enabled sandbox, you must sign out and sign back in before you can see the changes you deployed in the runtime.

You cannot publish the flexfield from the sandbox to the mainline. You must use the Define Flexfields task flow pages to deploy the flexfield for access by users of the mainline because the flexfield configuration in the mainline is the single source of truth.

**Deploying Flexfields Using the Command Line: Explained**

You can use the Manage Key Flexfields, Manage Descriptive Flexfields, and Manage Extensible Flexfields tasks to deploy flexfields. You can also use WebLogic Server Tool (WLST) commands for priming the Metadata Services (MDS) repository with predefined flexfield artifacts and for deploying flexfields. The table describes the available commands.

<table>
<thead>
<tr>
<th>WebLogic Server Tool Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>deployFlexForApp</td>
<td>Deploys all flexfields for the specified enterprise application. Only flexfields whose status is other than deployed are affected by this command unless the option is enabled to force all flexfields to be deployed regardless of deployment status. Initial application provisioning runs this command to prime the MDS repository with flexfield artifacts.</td>
</tr>
<tr>
<td>deployFlex</td>
<td>Deploy a single flexfield regardless of deployment status</td>
</tr>
<tr>
<td>deployPatchedFlex</td>
<td>Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.</td>
</tr>
</tbody>
</table>

Executing these commands outputs a report at the command line. The report provides the following information for every flexfield that is processed.

- Application identity (APPID)
- Flexfield code
- Deployment result, such as success or error

In case of errors, the report lists the usages for which the errors were encountered. If a runtime exception occurs, the output displays the traceback information.

Consider the following aspects of command line deployment.

- Preparing to use the WLST flexfield commands
- Using the `deployFlexForApp` command
- Using the `deployFlex` command
- Using the `deployPatchedFlex` command
- Exiting the WLST and checking the results

**Preparing To Use the WLST Flexfield Commands**

You can only execute the WLST flexfield commands on a WebLogic Administration Server for a domain that has a running instance of the Oracle Fusion Middleware Extensions for Applications (Applications Core) Setup application.

For more information on deploying the Applications Core Setup application, see the Oracle Fusion Applications Developer's Guide.

Ensure that the AppMasterDB data source is registered as a JDBC data source with the WebLogic Administration Server and points to the same database as the ApplicationDB data source.

Start the WebLogic Server Tool (WLST) tool, if it is not currently running.

UNIX:

```
sh $JDEV_HOME/oracle_common/common/bin/wlst.sh
```

Windows:

```
wlist.cmd
```

Connect to the server, replacing the user name and password arguments with your WebLogic Server user name and password.

```
connect('wls_username', 'wls_password', 'wls_uri')
```

The values must be wrapped in single-quotes. The `wls_uri` value is typically `T3://localhost:7101`.

For more information on the WLST scripting tool, see the Oracle Fusion Middleware Oracle WebLogic Scripting Tool.

**Using the `deployFlexForApp` Command**

The `deployFlexForApp` command translates the product application's predefined flexfield metadata into artifacts in the MDS repository.

**Important**

This command is run automatically when you provision applications. However, after custom applications development, you must run the `deployFlexForApp` command after you configure your application to read the flexfield artifacts from the MDS repository and before you log into the application for the first time, even if there is no predefined flexfield metadata.

This command does not deploy flexfields that have a status of Deployed unless the `force` parameter is set to `true` (the default setting is `false`).
For more information on priming the MDS partition with configured flexfield artifacts, see the Oracle Fusion Applications Developer's Guide.

From the WLST tool, execute the following commands to deploy the artifacts to the MDS partition, replacing `product_application_shortname` with the application's short name wrapped in single-quotes.

```
deployFlexForApp('product_application_shortname', 'force')
```

In a multi-tenant environment, replace `enterprise_id` with the Enterprise ID to which the flexfield is mapped. Otherwise, replace with 'None' or do not provide a second argument.

To deploy all flexfields regardless of their deployment status, set `force` to 'true' (the default setting is 'false'). If you want to deploy all flexfields in a single-tenant environment, you either can set `enterprise_id` to 'None', or you can use the following signature:

```
deployFlexForApp(applicationShortName='product_application_shortname',force='true')
```

**Tip**

The application's short name is the same as the application's module name.

For more information about working with application taxonomy, see the Oracle Fusion Applications Developer's Guide.

**Using the deployFlex Command**

From the WLST tool, execute the following command to deploy a flexfield, replacing `flex_code` with the code that identifies the flexfield, and replacing `flex_type` with the flexfield's type, which is either DFF, KFF, or EFF.

```
deployFlex('flex_code', 'flex_type')
```

The values must be wrapped in single-quotes.

**Using the deployPatchedFlex Command**

Use the `deployPatchedFlex` command for situations where the patching framework does not invoke the command, such as when an application has been patched offline.

If the installation is multi-tenant enabled, the command deploys all patched flexfields for all enterprises. This command is not intended to be invoked manually.

Check with your provisioning or patching team, or the task flows for managing flexfields, to verify that the flexfield has a Patched deployment status.

From the WLST tool, execute the following command to deploy the artifacts to the MDS partition.

```
deployPatchedFlex()
```

**Exiting the WLST and Checking the Results**

To exit the tool, execute the following command.
disconnect()

Optionally, sign into the application, access user interface pages that contain flexfields, and confirm the presence of flexfields for which configuration exists, such as value sets, segments, context, or structures.

Define Flexfields: Manage Value Sets

Value Sets: Explained

A value set is a set of valid values that you assign to a flexfield segment.

An end user enters a value into a flexfield segment while using the application. The flexfield validates the segment against the set of valid values that you configured as a value set and assigned to the segment.

For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

Caution

Be sure changes to a shared value set are compatible with all flexfields segments using the value set.

Defining value sets involves making decisions about the following.

- Validation
- Security
- Precision and scale
- Usage and deployment

Validation

The following types of validation are available for value sets.

- Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set does not present a list of valid values to users.
You can build a tree structure from the values in an independent value set whose data type is character.

**Note**

Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

For more information, see the Oracle Fusion Applications Extensibility Guide.

**Security**

Value set security only works in conjunction with usage within flexfield segments. If a value set is used standalone, meaning outside a flexfield, value set security is not applied, but Oracle Fusion data security is enforced.

You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

Value set security applies at the value set level. If a value set is secured, every usage of it in any flexfield is secured. It is not possible to disable security for individual usages of the same value set.

Value set security applies to independent, dependent or table-validated value sets.

Value set security applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security does not determine which descriptive flexfield data is shown upon querying.

Security conditions defined on value sets will always use table aliases. When filters are used, table aliases are always used by default. When predicates are defined for data security conditions, make sure that the predicates will also use table aliases.

For key flexfields, the attributes in the view object that correspond to the code combination ID (CCID), structure instance number (SIN) and data set number (DSN) cannot be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the CCID segment is the common attribute.

**Precision and Scale**

For a value set with the data type Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

**Usage and Deployment**

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.

The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.
For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment. Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

**Defining Value Sets: Critical Choices**

Validation and usage of value sets determine where and how end users access valid values for attributes represented by flexfield segments.

**Tip**

As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

**Value Sets for Context Segments**

When assigning a value set to a context segment, you can only use table-validated or independent value sets. The data type must be character and the
maximum length of the values being stored must not be larger than column length of the context.

**Format Only Validation**

The format only validation type enables end users to enter any value, as long as it meets your specified formatting rules. That is, the value must not exceed the maximum length you define for your value set, and it must meet any format requirements for that value set.

For example, if the value set allows only numeric characters, your user could enter the value 456 (for a value set with maximum length of three or more), but could not enter the value ABC. A format only value set does not otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how many digits should follow the radix separator.

**Interdependent Value Sets**

You cannot specify a dependent value set for a given segment without having first defined an independent value set that you apply to another segment in the same flexfield. You use a dependent value set to limit the list of values for a given segment based on the value that the end user has chosen for a related independent segment. The available values in a dependent list and the meaning of a given value depend on which value was selected for the independently validated segment.

For example, you could define an independent value set of U.S. states with values such as CA, NY, and so on. Then you define a dependent value set of U.S. cities, with values such as San Francisco and Los Angeles that are valid for the independent value CA, and New York City and Albany that are valid for the independent value NY. In the UI, only the valid cities can be selected for a given state.

Because you define a subset value set from an existing independent value set, you must define the independent value set first. End users do not need to choose a value for another segment first to have access to the subset value set.

**Table Validation**

Typically, you use a table-validated set when the values you want to use are already maintained in an application table (for example, a table of vendor names). Table validation allows you to enable a segment to depend upon multiple prior segments in the same context or structure.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it does not have bind variables. If it has bind variables, the assumption is that the values are unique in the value set.

**Range**

In the case of format, independent, or dependent value sets, you can specify a range to further limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of
segments where one segment represents the low end of the range and another segment represents the high end of the range

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100. If you use a table value set, you cannot reference flexfield segments in the WHERE clause of the value set. For example, the WHERE clause cannot reference a segment or a value set.

**Security**

In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

When you enable security on a table-validated value sets, the security rule that is defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set should not assume anything about the bind variables; it must assume the whole list of values is available and write the rule, for example, to allow x, or to allow y and z. By default in data security all values are denied, and show only rows to which access has been provided.

**Maintenance**

There is no need to define or maintain values for a table-validated or subset value set, as the values are managed as part of the referenced table or independent value set, respectively.

If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select **Translatable**. This does not require you to provide translated values now, but you cannot change this option if you decide to provide them later.

For more information about defining value sets, see the Oracle Fusion Applications Extensibility Guide.

**Define Flexfields: Manage Descriptive Flexfields**

**Descriptive Flexfields: Explained**

Descriptive flexfields provide a way to add custom attributes to entities, and define validation and display properties for them. These attributes are generally standalone. They don’t necessarily have anything to do with each other and are not treated together as a combination.

All Oracle Fusion Applications business entities that you can access are enabled for descriptive flexfields. Descriptive flexfields are entirely optional. You can choose to configure and expose the descriptive flexfield defined and registered in your database, or not.
A descriptive flexfield provides a set amount of segments for an entity. The segments of a descriptive flexfield are made available to end users as individual fields in the application user interface.

**Segments and Contexts**

Descriptive flexfield segments are of the following types.

<table>
<thead>
<tr>
<th>Segment Type</th>
<th>Runtime Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global segment</td>
<td>Always available</td>
</tr>
<tr>
<td>Context segment</td>
<td>Determines which context-sensitive segments are displayed</td>
</tr>
<tr>
<td>Context-sensitive segment</td>
<td>Displayed depending on the value of the context segment</td>
</tr>
</tbody>
</table>

In the figure, a descriptive flexfield has one context segment called Category for which there are three values: Resistor, Battery, and Capacitor. In addition, the descriptive flexfield consists of two global segments that appear in each of the contexts, and three context-sensitive segments that only appear in the context in which they are configured.

Application development determines the number of segments available for configuring. During implementation you determine which attributes to add using the available segments, and the context values and the combination of
attributes in each context. A segment can be used for different attributes, such as Height in Context1 and Color in Context2. Each segment of a descriptive flexfield that you make available to end users is exposed in the user interface as an individual field.

**Value Sets**

For each global and context-sensitive segment, you configure the values allowed for the segment and how the values that end users enter are validated, including interdependent validation among the segments.

**Managing Descriptive Flexfields: Points To Consider**

Configuring descriptive flexfields involves managing the available flexfields registered with your Oracle Fusion Applications database and configuring their flexfield-level properties, defining and managing descriptive flexfield contexts, and configuring global and context-sensitive segments.

Every descriptive flexfield is registered to include a context segment, which you may choose to use or not.

**Segments**

Assign sequence order numbers to global segments. Segment display is always in a fixed order. You cannot enter a number for one segment that is already in use for a different segment.

Value sets are optional for context segments. The value set that you specify for a context segment consists of a set of context codes, each of which corresponds to a context that is appropriate for the descriptive flexfield. If you do not specify a value set for a context segment, the valid values for that context segment are derived from the context codes. The definition of each context segment specifies the set of context-sensitive segments that can be presented when that context code is selected by the end user.

For reasons of data integrity, you cannot delete an existing context. Instead, you can disable the associated context value in its own value set by setting its end date to a date in the past.

You can configure the individual global segments and context-sensitive segments in a descriptive flexfield. These segment types are differentiated by their usage, but they are configured on application pages that use most of the same properties.

**Usages**

Descriptive flexfield usages allow for the same definition to be applied to multiple entities. Descriptive flexfield tables define the placeholder entity where the flexfield segment values are stored once you have configured the descriptive flexfield.

**Parameters**

Parameters are public arguments to a descriptive flexfield. Parameters provide outside values in descriptive flexfield validation. Parameters can be referenced
by the logic that derives the default segment value and in table-validated
VALUE set WHERE clauses.

**Delimiters**

A segment delimiter or separator visually separates segment values when the
flexfield is displayed as a string of concatenated segments.

**Business Intelligence**

If a descriptive flexfield is registered in the database as enabled for Oracle
Business Intelligence, you can specify that a global, context, or context-sensitive
segment is BI Enabled so it is available for use in Oracle Business Intelligence.
The BI Enabled setting is otherwise unavailable.

If you BI-enable a context-sensitive segment, the business components of the
flexfield are flattened automatically to include a separate attribute for each of
the BI-enabled segments, even if some of them serve the same purpose in your
application. If you BI-enable a global segment or context segment, the business
components of the flexfield automatically include a single attribute for the global
segment or context segment. Descriptive flexfields do not support equalization
across context-sensitive segments.

**Define Flexfields: Manage Extensible Flexfields**

**Extensible Flexfields: Explained**

Extensible flexfields are like descriptive flexfields, with some additional features.

- You can add as many context-sensitive segments to the flexfield as you
  need. You are not dependent on the number of segments predefined and
  registered for the flexfield.

- You can configure a one-to-many relationship between the entity and its
  extended attribute rows.
  - A row of data can have multiple contexts associated with it.
  - A row of data can have multiple occurrences of the same context.

- You can configure contexts in groups so the attributes in the context
  always appear together in the user interface.

- You can use existing hierarchical categories so that entities inherit the
  contexts that are configured for their parents. Contexts are reusable
  throughout categories.

When you configure a context for multiple rows per entity, the segments are
displayed as a table.

Unlike descriptive flexfields, the extension columns corresponding to extensible
flexfields segments are part of extension tables, separate from the base
application table. Unlike descriptive flexfield contexts, the set of attributes in
an extensible flexfield context remains constant and does not differ by context
value.
An extensible flexfield describes an application entity, with the runtime ability to expand the database that implementation consultants can use to define the data structure that appears in the application.

Extensible flexfields support one-to-many relationships between the entity and the extended attribute rows.

**Usages**

As with descriptive flexfields, you can define multiple usages for an extensible flexfield, which enables several application tables to share the same flexfield. For example, a flexfield for shipping options can be used by both a Supplier table and a Buyer table. In addition, you can associate a context with one, some, or all of the flexfield's usages. Thus, with the shipping information example, you can associate a warehouse context with the Supplier usage, a delivery location context with the Buyer usage, and a ship-via context with all usages.

**Categories**

You can define categories for extensible flexfields, and you can associate any combination of contexts with a given category. For example, the Electronics and Computers category hierarchy might include a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on. The Home Entertainment product might have contexts that specify voltage, dimensions, inputs and outputs. Contexts are reusable within a given extensible flexfield. For example, the dimensions context could be assigned to any category that needs to include dimensional information.

**Pages**

Extensible flexfields allow you to combine contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

Each application page corresponds to one extensible flexfield category, with a separate region of the page for each associated context.

**Managing Extensible Flexfields: Points To Consider**

Configuring extensible flexfields involves managing the available flexfields registered with your application database and configuring their flexfield-level properties, defining contexts, categories, and pages, and configuring the segments for each extensible flexfield.

**Contexts**

A context can be defined as single row or multi row. Single row contexts are the same as descriptive flexfields contexts. A single row context has only one set of context-sensitive segments. A multi-row context enables you to associate multiple sets of values with the same object instance.
For example, for a BOOK table, you could create a multi-row context named chapters that contains a chapter segment and a number of pages segment. Multiple chapters can then be associated with each book in the BOOK table.

Set the context to translatable so free-form text entered by end users is stored in the language of the user’s locale, and different translations of that text can be stored in other languages. Segments in the translated contexts should utilize format-only value sets for storing free-form, user-entered text.

Categories

A category is a grouping of related data items that can be considered to belong together. You can associate any combination of contexts with a given category.

A category hierarchy logically organizes a set of categories.

For example, the Electronics and Computers category hierarchy might include a Computer category and a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on.

A category can be a child or sibling of an existing category. The hierarchy can be as simple or as complex as desired, with any combination of zero or more sibling categories and zero or more child categories. If no category is defined, the data items are grouped under a single predefined default category.

Each category has associated contexts that store relevant information about a data item in that category. For example, a Home Entertainment product has contexts that specify Voltage, Dimensions, Inputs and Outputs. Contexts are reusable within a given extensible flexfield; the Dimensions context could be assigned to any category that needs to include dimensional information.

If a hierarchy includes child categories, each child category inherits the contexts from its parent category; for example, the Home Entertainment category inherits Voltage and Dimensions from the Electronics and Computers category.

Each extensible flexfield is associated with a particular category hierarchy. You can think of category hierarchies as the defining framework for extensible flexfields and their contexts. A category hierarchy specifies which contexts are valid for each category.

An extensible flexfield can include multiple contexts which you define to support a given category. These contexts can be suitable for a variety of different purposes, but within a particular category, some contexts might be considered to be related to, or dependent on, each other. You can combine these contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

For example, the Home Entertainment category might have an Electrical Specifications page that contains the Voltage, Inputs and Outputs contexts, and a Physical Specifications page that contains the Dimensions and Form Factor contexts.

Indexed Segments

You can designate an extensible flexfield segment as indexed so that it is one of the selectively required attributes an end user can use in an attribute search. If
you indicate in the Manage Extensible Flexfield UI page that a segment should be indexed, the column representing the segment must be added to the database index. This is commonly done by a database administrator (DBA).

When an extensible flexfield with indexed segments is deployed, search task flows are generated along with the other flexfield artifacts and specify the indexed attributes as selectively required. In the deployed extensible flexfield’s search task flow, an end user must specify at least one of the indexed attributes in the search criteria. This prevents non-selective searches which could cause performance issues.

For example, if you index the memory and processor attributes and ensure that the corresponding columns in the database are indexed, an end user can search an item catalog for computers by entering processor or memory or both as a search criterion. No search is performed if an end user enters an attribute as search criteria that is not indexed.

**Define Flexfields: Manage Key Flexfields**

**Key Flexfields: Explained**

Key flexfields provide a means to capture a key such as a part number, a job code, or an account code. A key flexfield consists of one or more segments, where each segment can have a meaning.

For example, a part number 10-PEN-BLA-450 might correspond to a black pen from vendor #450 sold by division #10 (office supplies). Behind the scenes, the application uses a unique number, 13452, for this part, but the end user always see the 10-PEN-BLA-450 part number.

The following aspects are important to understanding key flexfields.

- Architecture
- Segments and segment labels
- Structures
- Segment and structure instances
- Combinations
- Dynamic combination creation
- Security

Key flexfields are not optional. You must configure key flexfields to ensure that your applications operate correctly. You configure and maintain key flexfield definitions with the Manage Key Flexfields task.

**Architecture**

When you configure a key flexfield, you define metadata about the key flexfield such as how many segments are in a structure, how many structures the flexfield
uses, what value sets each segment uses, and so on. This is flexfield metadata stored in flexfield metadata tables.

Based on the flexfield metadata, actual part numbers are captured at runtime as a combination of segment values and stored in a combinations table. A combinations table contains all the segment columns for a flexfield, plus a unique ID column and a structure instance number column that differentiates multiple arrangements of the segment columns.

For example, a part number that can be comprised of multiple segments can be represented by a key flexfield. A part number key flexfield has a corresponding combinations table, where the flexfield stores a list of the complete codes, with one column for each segment of the code, together with the corresponding unique ID and structure instance number for the code. When users define a new part number or maintain existing part numbers in the parts catalog, they directly maintain rows in the combination table.

The foreign key table contains a different business entity than the combinations table. For example, the business entity in the foreign key table is order lines or invoice lines that contain foreign key references to parts for ordering and so on. Any number of foreign key tables can reference a particular entity represented by a key flexfield.

**Segments and Segment Labels**

A key flexfield consists of segments. Segments consist of a prompt, a short prompt, display width, a number that determines where in the sequence of a key flexfield structure the segment exists, the range type and the column name of the attribute being captured by the segment, a default value set and a label for the segment. A segment label identifies a particular segment of a key flexfield. Segment labels are defined and made available by applications development.

Applications identify a particular segment for some purpose such as security or computations. Segment name or segment order cannot reliably identify a segment because key flexfield segments can be configured to appear in any order with any prompts. A segment label functions as a tag for a segment.

For example, Oracle Fusion General Ledger needs to identify which segment in the Accounting Flexfield contains balancing information and which segment contains natural account information. General Ledger uses a segment label to determine which segment you are using for natural account information. When you define your Accounting Flexfield, you must specify which segment label apply to which segments.

Some labels must be unique, and cannot be applied to more than one segment in each structure. Other labels are required, and must be applied to at least one segment in each structure.

A segment label orients an end user’s search of segments, such as the Cost Center label for all segments across key flexfields that capture a value for cost center.

**Structures**

A key flexfield structure definition includes the number of segments and their order.
A delimiter separates the segments when they appear to end users. The delimiter value of a structure specifies the character used to visually separate segment values when the key flexfield is displayed as a string of concatenated segments in the UI.

Tip

Choose the delimiter value of your key flexfield carefully so that it does not conflict with the flexfield data. For example, if your data frequently contains periods, such as in monetary or numeric values, do not use a period as your segment separator. Any character you expect to appear frequently in your segment values or descriptions is not a good choice for the delimiter.

If you change the configuration of a key flexfield, such as the delimiter, the change affects the previously stored key flexfields with that structure.

Each structure can have one or more segments. Thus a segment is a child of a structure. If you want to store a particular segment, such as Cost Center, in two different structures, you must define the segment separately in each structures.

Structure and Segment Instances

You can define multiple configurations of a key flexfield structure. These structure instances have the same segment structure, in the same sequence order. They differ primarily in how each segment is validated. You define a structure instance for each key flexfield and each key flexfield structure instance.

The segments in a key flexfield structure instance are segment instances. A segment instance is a segment with a specific value set assigned to it.

If a key flexfield has been registered with a tree structure, you can specify a tree code for a segment instance, where the tree code defines a hierarchical relationship between the segment values.

Combinations

A combination is a complete code, or combination of segment values that makes up the code, that uniquely identifies an object.

For example, each part number is a single combination, such as PAD-YEL-11x14 or 01-COM-876-7BG-LTN. In these combinations, the hyphen is the segment separator. If you had ten parts you would define ten combinations. A valid combination is simply an existing or new combination that can currently be used because it is not out of date or disabled, and does not violate cross-validation or security rules. A combination has different segments depending on the flexfield structure being used for that combination. Any combination is associated with only one particular flexfield structure.

Many Oracle Fusion Applications products refer to a key flexfield combination by using the name of the entity or the key flexfield itself. For example, Oracle Fusion Assets uses the asset key flexfield and refers to one of its combinations as an asset key or asset key flexfield. In another example, other Oracle Fusion Applications products including Oracle Fusion General Ledger (GL) refer to combinations of the accounting flexfield as account or GL account.
Each key flexfield has one corresponding table, known as the combinations table, where the flexfield stores a list of the complete codes, with one column for each segment of the code, together with the corresponding unique ID number (a code combination ID number or CCID) for that code. Then, other tables in the application have a column that stores just the unique ID for the code. For example, you may have a part number code, such as PAD-YEL-11x14. The Parts combinations table stores that code along with its ID, 57494. If your application allows you to take orders for parts, you might then have an Orders table that stores orders for parts. That Orders table would contain a single column that contains the part ID, 57494, instead of several columns for the complete code PAD-YEL-11x14.

Typically one combinations page maintains the key flexfield, where the key flexfield is the representation of an entity in your application. The combinations page is where you maintain individual combinations, such as part numbers.

**Dynamic Combination Creation**

Dynamic combination creation is the insertion of a new valid combination into a combinations table from a page other than the combinations page.

Dynamic combination creation may be enabled at the following levels.

<table>
<thead>
<tr>
<th>Level Of Dynamic Combination Creation</th>
<th>Controlled By:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexfield</td>
<td>Application development</td>
</tr>
<tr>
<td>Each usage or reference to the key flexfield</td>
<td>Application development</td>
</tr>
<tr>
<td>Structure instance</td>
<td>Administrators and implementation consultants</td>
</tr>
<tr>
<td>Other</td>
<td>Administrators and implementation consultants</td>
</tr>
</tbody>
</table>

If your key flexfield or certain usages or references of the key flexfield do not permit dynamic combination creation, you may control whether dynamic combination creation is enabled for each structure instance. If enabled, a user can enter a new combination of segment values using the flexfield window from a foreign key page. For example, when entering a transaction, a GL user can enter a new expense account code combination for an account that does not yet exist. Your application creates the new account by inserting the new combination into the combinations table behind the scenes. Assuming that the new combination satisfies any existing cross-validation rules, the flexfield inserts the new combination into the combinations table, even though the combinations table is not the underlying table for the foreign key page.

**Managing Key Flexfields: Points to Consider**

Consider the plans for a key flexfield, security, and resulting runtime pages when configuring key flexfields.

**Planning**

Plan structures carefully and allow for future needs.
Caution

Do not change the number, order, and maximum length of segments once you have acquired flexfield data.

Business Intelligence

If a key flexfield is registered in the database as enabled for Oracle Business Intelligence integration, you can specify that a segment instance is BI Enabled, which makes the segment instance available for use in Oracle Business Intelligence. The BI Enabled setting is otherwise unavailable.

If you BI-enable multiple segment instances from the same key flexfield, the business components of the flexfield are flattened automatically to include a separate attribute for each of the BI enabled segment instances, even if some of them serve the same purpose in your application. You can prevent this duplication and the extra workload and complexity that result, by enabling equalization for any set of segment instances that serve the same purpose in different structure instances. Enable equalization by assigning a unique segment label to those segment instances that you wish to be equalized across structure instances. Equalized segment instances appear in the flattened flexfield as a single attribute. Segment instances that are assigned non-unique segment labels are not equalized.

Security

Oracle Fusion data security enforces value set security.

Within key flexfields, value set security applies to the selection of the individual segment values in the segment list of values. When selecting a key flexfield segment value from the combination table, data security allows display of only the combinations whose segment values you have access to. Applications development controls whether or not value set security rules propagate to the foreign key table. By default they do.

Runtime Pages

Application development determines the user interface (UI) pages used to render flexfields. The types of key flexfield UI pages are as follows.

- Combinations pages where underlying entity objects use the combinations table itself
- Foreign key pages where the underlying entity objects contain a foreign key reference to the combinations table
- Partial usage page where some or all of the key flexfield’s segment columns are in a product table

The same key flexfield can be used in different ways on different pages.

A page with a foreign key reference has a base table or view that contains a foreign key reference to a combinations table with the actual flexfield segment columns. This allows manipulating rows containing code combination IDs (CCID).
A page with partial usage of a key flexfield presents segments that are defined on a product's transactional table in addition to being defined on a combinations table. In the case of a partial usage page, it is possible that only part of the configuration is visible. This allows the key flexfield to behave more like a descriptive flexfield.

A code combination maintenance page or combinations page presents the combinations table. This allows directly creating and maintaining code combinations. The combinations table contains all key flexfield segment columns and a unique ID column.

A typical application has one and only one combinations page. An application might not have a combinations page if it does not support maintenance by administrators.

A page containing a search region enables end users to select which attributes of the key flexfield view object to use as criteria to search for flexfield metadata.

For example, you can configure seven segments for the Account key flexfield. In a foreign key reference page, end users see the typical key flexfield picker with all seven segments where they can search for combinations. In a partial usage page using the same key flexfield, end users potentially could see only a single segment such as the Cost Center labeled segment, or they might see multiple segments but displayed as individual segments rather than as a picker for choosing combinations.

For more information on key flexfield pages, see the Oracle Fusion Applications Developer's Guide.

**Key Flexfield Structures: Explained**

A key flexfield structure arranges the segments of a key so you can reuse a single key flexfield in multiple combinations of the same or a subset of segments. Multiple instances of a single structure can accommodate differences in the value sets assigned to the structure's segments.

The structure determines the following aspects of a key flexfield.

- The segments to include
- The order of the segments
- Segment labels on the included segments
- Properties for each segment applied to the instances of the segments in an instance of the structure

**Managing Key Flexfield Structures**

All the segments defined for a key flexfield are available to be included in a key flexfield structure.
You can define as many segments as there are defined segment columns in your key flexfield combinations table.

**Restriction**

Be sure to add segments in the order that your key requires. Once deployed, the order cannot be changed.

Enable segments to indicate that they are in use. A flexfield does not display disabled segments in runtime.

**Tip**

To protect the integrity of your data, disable a segment if you have already used it to enter data.

---

**Key Flexfield Structure Instances and Segment Instances: Explained**

A key flexfield structure can have one or more alternate structure instances.

The instances of a key flexfield structure share the following aspects of the structure.

- The same set of segments
- The same arrangement of segments
- The same properties at the segment and structure levels

Differences among structure instances at the structure level include whether dynamic combination creation is allowed.

Differences among segment instances at the structure instance level include the following.

- Value set
- Default type and default value
- Tree code
- Whether the segment is any of the following
  - Required
  - Displayed
  - Enabled for business intelligence
  - Optional or required as a query criterion

For example, you could use one group of value sets for the US and another for France.
The figure shows two structures instances for a part number structure. The structures differ in the number of segments and the segment separators used. The structure instances of a structure share all properties that are defined for the structure, but can vary in the properties defined at the structure instance or segment instance level, such as the value set assigned to the segment instances.

Query Required Segment Instances

You can designate a key flexfield segment instance as query required so that it is one of the selectively required attributes an end user can use in a key flexfield combination search. If you indicate in the Manage Key Flexfields UI page that a segment instance should be indexed, the column representing the segment must be added to the database index. This is commonly done by a database administrator (DBA).

Following deployment, the combination picker of the key flexfield displays the query required attributes as selectively required. An end user must specify at
least one of the query required attributes in the search criteria. This prevents non-selective searches that could cause performance issues.

For example, if you mark the cost center and account attributes as query required and ensure that the corresponding columns in the database are indexed, an end user can search for combinations by entering cost center or account or both as a search criterion. No search is performed if an end user does not enter at least one query required attribute as search criteria.

Tip
Index the Structure Instance Number column on your combinations table to improve performance.

Dynamic Combinations

If a key flexfield supports dynamic combination creation, you can choose to enable this feature by selecting Dynamic Combination Creation Allowed. This will allow end users to enter values at runtime that produce new code combinations for the flexfield. If not enabled, new valid combinations can only be entered using the combinations table for the flexfield.

Trees

If a tree code has been defined for the value set assigned to the segment instance, and you assign the tree code to the segment instance, tree hierarchy search operations are available on the segment values.

For a segment instance to be based on a tree, the following must be true.

- Application development registered the key flexfield with a tree structure.
- A tree code for that tree structure exists.
- The tree code that includes tree versions containing the values of the value set assigned to the segment instance.
- You assign the desired tree code directly to the segment instance.

Provided these conditions are satisfied, different segment instances that use the same value set can be assigned the same or different tree codes, meaning they use a different hierarchy definition over the same values.

Key Flexfields: Example

A key flexfield can capture expense account information.

Scenario

When entering details for each expense, the user specifies an account to which the expense is charged.
Entering Expense Accounts

A user interface for entering expenses gives the user the option of selecting an expense account that identifies the cost center and other details needed for processing the expense.

Analysis

The expense account field is a foreign key reference to a code combination (EXPENSE_LINES.EXPENSE_ACCOUNT = ACCOUNTS.CCID).

Code Combination Table for Entering Accounts and Employees

The code combination table supports entering account information, such as for expense accounts.

The figure shows the origin in the code combination table of the account specified by the user. The code combination ID record stores the information of the key flexfield segments used to assemble the expense account based on the key flexfield configuration.
The combinations page, which is the maintenance page for the key flexfield, is for managing rows in the combination table. In this example, managing the combinations means adding or editing account numbers that adhere to the key flexfield metadata rules.

The figure shows the code combination details for the example expense account reflected in the flexfield configuration and the code combination table.

If dynamic combination creation is not enabled, then when entering an expense line, the user can only select an account that already exists in the ACCOUNTS (combinations) table. If they require an account that does not exist, they must consult with the appropriate application administrator who can add the account to the combinations table.

If dynamic combination creation is enabled, then when entering an expense line, the user can either select a pre-existing account, or type in a new account that created dynamically on the fly in the ACCOUNTS (combinations) table. Once the new combination is created, the same user can refer to it on the expense line.

When managing employee information, the user specifies the cost center that the employee belongs to. The cost center field corresponds to a single, labeled segment of the Account Key Flexfield and has metadata defined such as the allowable value set for that segment.

In this figure, instead of specifying a cost center ID reference to an account, only the Cost Center segment is used and the value is stored directly on the employee table.
FAQs for Define Flexfields

Why did my flexfield changes not appear in the runtime UI?

The ADF business components or artifacts of a flexfield, which are generated into an metadata services (MDS) repository when the flexfield is deployed, are cached within a user session. You must sign out and back in again to view flexfield definition changes reflected in the runtime application user interface page.

A flexfield’s status relative to its deployment determines whether the flexfield segments as currently defined in the metadata are available to end users. The flexfield segments seen by end users in the runtime correspond to the flexfield definition that was last deployed successfully.

What happens if a value set is security enabled?

Value set security is a feature that enables you to secure access to value set values based on the end user’s role in the system.

As an example, suppose you have a value set of US state names. When this value set is used to validate a flexfield segment, and users can select a value for the segment, you can use value set security to restrict them to selecting only a certain state or subset of states based on their assigned roles in the system.

For example, Western-region employees may choose only California, Nevada, Oregon, and so on as valid values. They cannot select non-Western-region states. Eastern-region employees may choose only New York, New Jersey, Virginia, and so on as valid values, but cannot select non-Eastern-region states. Value set security is implemented using Oracle Fusion Applications data security.

How can I set a default value for a flexfield segment?

When you define or edit a flexfield segment, you specify a default value from the values provided by the value set assigned to that segment.
You can set the default value to be a parameter, which means the entity object attribute to which the parameter you choose is mapped will provide the initial default value for the segment.

You can set to be a constant, if appropriate to the data type of the value set assigned to the segment.

In addition to an initial default value, you can set a derivation value for updating the attribute’s value every time the parameter value changes. The parameter you choose identifies the entity object source attribute. Any changes in the value of the source attribute during runtime are reflected in the value of the segment.

If the display type of the segment is a check box, you can set whether the default value of the segment is checked or unchecked.

**Define Attachments**

**Attachments: Explained**

Attachments are pieces of supplementary information that users can associate with specific business objects such as expense reports or purchase orders. Attachments can be URLs, desktop files, text, or in cases where available, repository folders. For any given business object, a user may be able to only view attachments, or also create, delete, or edit attachments, depending on security. For more information on an introduction to attachments, see the Oracle Fusion Applications Developer’s Guide.

**Repository**

Attachments are stored in a content server, the Oracle Enterprise Content Management document repository. Users have no real interaction with the repository unless the repository mode is enabled for attachments on specific business objects. In which case, users can share attachments among objects, update attachments by checking them out of and back in to the repository, and perform other tasks. Access to attachment files is controlled by a digital signing mechanism. Depending on security, users might have direct access to the repository.

**Security**

Data security that applies to a specific business object also applies to attachments for that object, as determined by the attachment entity defined for the object. For example, if a user has no access to a specific expense report, then the same user cannot access attachments for the expense report. You can also use attachment categories to control access and actions on attachments, based on roles associated with the category. For more information on securing attachments, see the Oracle Fusion Applications Developer’s Guide.

**Attachment Entities: Explained**

An attachment entity is usually a database entity, for example a table or view, that represents a business object attachments can be associated with. Each
attachment UI must be defined with a corresponding attachment entity, which not only identifies the business object to attach to, but also controls what users can do. Attachment entities are used only in the context of attachments and exist separately from the database entities that they are based on.

Edit and create attachment entities on the Manage Attachment Entities page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Attachment Entities task. Though you would generally use predefined attachment entities with attachment UIs, you might need to create new entities, for example when developing custom UIs.

**Entity Names**

An attachment entity name should match the name of the table or view that represents the business object to attach to. The name is also used in the repository folder that is automatically created to store attachments for the entity. The attachment entity display name should be something that users know to represent the business object.

**Database Resource**

The data security policies associated with the database resource defined for the attachment entity would apply to attachments for that entity. For example, based on the database resource for the expense reports attachment entity, the same policies apply to attachments for expense reports. The database resource value must match the value in the OBJ_NAME column in the FND_OBJECTS table for the business object that the entity represents.

**Enabling Security**

Security based on the database resource associated with the attachment entity is always in effect. What you can enable or disable is security based on attachment categories. If any of the attachment categories associated with the attachment entity has data security defined, then that security applies to this entity only if enabled.

**Attachment Entities and Attachment Categories: How They Work Together**

The association between attachment entities and categories determines which categories can be used for an entity. For example, categories associated with the expense report attachment entity are available to be implemented in attachment UIs for expense reports. You can define these associations when managing either entities or categories. Any association changes in either the Manage Attachment Entities or Manage Attachment Categories page are reflected on the other page. You can access either page by starting in the Setup and Maintenance Overview page and searching for attachment tasks.

**Managing Entities**

You determine which attachment categories are relevant to a particular entity on the Manage Attachment Entities page, and each entity must have at least one category. Depending on configuration, any or all of the available categories for
that entity are used. For example, you assign three categories to the expense reports attachment entity. For a particular expense report page with attachments functionality, you can customize the attachments component to specify which of the three categories are used. Based on your selection, the data security defined for each category, if any, is applied to attachments on that page if the attachment entity has category-based security enabled.

Managing Categories

If you create an attachment category and need to assign it to multiple attachment entities, use the Manage Attachment Categories page. The association means the same as the association on the Manage Attachment Entities page.

FAQs for Define Attachments

What's an attachment category?

An attachment category is used to classify and secure attachments. Each attachment user interface must be defined with at least one category for users to be able to add attachments. If there are multiple categories, users can view them and select one when adding attachments. For example, attachments for an expense report can be categorized as receipts, scanned invoice images, and so on.

You can also associate roles with categories to determine user access and actions for attachments, based on the categories assigned to the attachment entity. For example, security for expense report attachments can be based in part on the categories assigned to the expense report attachment entity. You can define multiple categories per module, and add and manage custom categories for your own purposes. For more information on attachment category data security, see the Oracle Fusion Applications Developer's Guide.

Use the Manage Attachment Categories page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Attachment Categories task.

Set Activity Stream Options

Activity Stream Options: Highlights

Activity Stream is a region on the Oracle Fusion Applications Welcome dashboard and other pages in various applications. Users track the activities and transactions of other users in this region. You can set options that affect the all Activity Stream regions for all users across your site. Individual users can still override your settings through Activity Stream preferences.

Activity stream settings are described in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces. When you read content from that guide, note that:

- Your setup applies to all users, not just yourself or any individual user.
- You can disregard discussions about how to access the settings, because you access the Set Activity Stream Options page by starting in the Setup
and Maintenance Overview page and searching for the Set Activity Stream Options task.

**Setting Activity Stream Options**

- Define the types of users to display activities about in the Activity Stream region, the types of activities to track, and other settings.

See: Setting Activity Stream Preferences

**Manage Menu Customizations**

**Managing Menu Customizations: Highlights**

You can customize the Navigator menu, which is the main menu of Oracle Fusion Applications and is always available in the global area. Use the Manage Menu Customizations page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Menu Customization task.

An overview of customizing the Navigator menu is provided in the Oracle Fusion Applications Extensibility Guide. Details of the values you enter to define menu items are provided in the Oracle Fusion Applications Developer's Guide.

**Overview**

- Navigator menu customization involves managing items, which are nodes in the menu that take the user to the desired destination, and groups, which are categories of items. The label is the menu item or group text displayed to users, and only rendered items and groups are visible to users. Items and groups that are not rendered are displayed in italics in the menu customization pages. Refer to the Oracle Fusion Applications Extensibility Guide.

See: Customizing the Navigator Menu

**Menu Items**

When you customize items in the Navigator menu, you determine if the item takes users to a specific page in Oracle Fusion Applications or to another application or Web site. The details of what you enter to define the menu item depend on the type of destination. If you duplicate a menu item, the new item appears below the selected source item.

- For menu items to an Oracle Fusion Applications page, do not enter anything in the Destination field, but specify the focus view ID of the target page, Web application, secured resource name, application stripe, and page parameters list. The secured resource name identifies the page definition file that is used to secure resource grants for the page, for example oracle.apps.view.pageDefs.CaseList_Form_Attach_UIShellPagePageDef. The application stripe is the partition for the application in the policy store, for example crm. If you enter both a secured resource name and an application stripe, then the menu item is rendered only if the logged-in user has permission to view that secured resource. If either value is missing, then the item is not secured. Refer to the Oracle Fusion Applications Developer's Guide.
Define Common Applications Configuration for Marketing

See: Menu Attributes Added by Oracle Fusion Middleware Extensions for Applications

- For menu items to another application or Web site, enter only the full URL in the Destination field.

Manage Applications Core Common Reference Objects

Common Messages: Points to Consider

Common messages, which have message names that begin with FND_CMN and message numbers between 0 and 999, are used throughout Oracle Fusion Applications. Each common message can appear in multiple places in any product family. For example, the FND_CMN_NEW_SRCH message can be used for any search to indicate that no results were found. Common messages that are of type error or warning are part of the message dictionary.

Editing Common Messages

Because a common message can be used in any application, consider the ramifications if you edit any aspect of the message, including incident and logging settings. Changes would be reflected in all instances where the message is used. For example, if you change the message text, make sure that the text would make sense to all users across Oracle Fusion Applications who might see it.

Creating Common Messages

You can create custom common messages for use in multiple places within a single product. Do not begin the message name with FND_CMN, but use another suitable convention. The message number should be within the range that is designated for the product.

Document Sequences: Explained

In Oracle Fusion Applications, each business document or business event is uniquely identified by a document sequence number that you assign to it. However, the document sequencing feature must be turned on (enabled) on the business document or event to allow the assignment. For example, if document sequencing is enabled, you can assign a document sequence number to an invoice that gets generated in response to a purchase order. You can use document sequences as a proof to track successfully executed transactions as well as failed transactions. Additionally, a document sequence helps in generating an audit trail, which can be used to identify how a particular transaction passed through various applications.

Document sequencing can be managed automatically, manually, and gaplessly.

Note

Plan your document sequencing carefully before you use the options available in the application to apply sequence numbers. Avoid changes to the options.
after you saved your work on the Manage Document Sequences and Manage Document Sequence Categories pages.

**Automatic Sequencing**

Automatic document sequencing assigns a unique number to each document as it is generated, and this unique number is stored in the database. The numbering is sequential by date and time of creation. If you define a sequence to automatically number documents, you can provide an initial value to begin the sequence. In absence of a custom value, the default value 1 is used.

**Manual Sequencing**

Manual sequencing requires you to assign a unique number to each document before it is generated. In manual sequencing, the numerical ordering and completeness of a transaction is not enforced. Users can skip or omit numbers when entering the sequence value. However, each time that a number is assigned, the application validates its uniqueness.

**Gapless Sequencing**

Gapless sequencing is similar to automatic sequencing. It automatically generates a unique number for each document, but does that only for successfully generated documents. As a result, the sequence is maintained for all the documents that are generated, and no sequence numbers are lost due to incomplete or failed document generation.

**Important**

Use this type of sequencing only if necessary because it may affect the performance of the system and slow down transaction processing.

**Sequential Numbering Enforced Profile Options**

The Sequential Numbering Enforced profile validates the documents being created and applies the relevant document sequence, based on the selected option. The following options are available:

- **Always Used**: Sequential numbering is enforced for all document categories. When this option is set, users cannot create a document if its document category is not assigned any active document sequence. The system displays an error.

- **Partially Used**: Sequential numbering is enforced for all document categories. However, if no sequence is available for the document, the system displays a warning. Users can either proceed without a document sequence or create and assign a document sequence before proceeding.

- **Not Used**: Sequential numbering is not enforced. In this mode, although the documents created do not require a document sequence to be assigned, the system checks for the uniqueness of the document number provided. The system does not display any warning or error.

**Restriction**
At the site level, the profile is by default set to Partially Used. Avoid changing this option as doing so may interfere with the validation logic and the data that was stored using the earlier option may not appear. However, you can change the option at the product or user level.

**FAQs for Manage Applications Core Common Reference Objects**

**How can I determine who sees the user or administrator components of a message?**

Set the Message Mode profile option to determine if the administrator or user details and action components of the message text is displayed. For example, you can set the profile option at the user level for a particular user to see the administrator information. All users can see the message short text and cause; the profile option does not apply to those message components. This profile option also applies only to messages in the message dictionary.

**Define WebLogic Communication Services Configuration**

**Click to Dial: Top Tasks**

Click to Dial is a feature of the customer contact process. Phone communication to customers and employees is initiated with a click of the mouse, leveraging your customer contact information and the application context. The Click to Dial feature uses Oracle WebLogic Communication Services, OWLCS, to enable communications. Applications that provide the Click to Dial functionality do so primarily through contextual actions in the application.

Additionally, Click to Dial utilizes Oracle Fusion Interactions as an optional transaction logging feature that will track information about the call such as the customer, call participants, a timestamp noting the start time of the call, the direction of the communication, in or outbound, and the resolution code.

Click to Dial integrates with your telephony environment and must be manually enabled in your deployment. This topic highlights what is required to set up the Click to Dial feature and to implement logging of the calls made using the Click to Dial feature.

**Terms used in setting up these communications**

- **PSTN**: Public switched telephone network is the network of the world’s public circuit-switched telephone networks.
- **SIP**: Session initiation protocol, an open signaling protocol standard that is used to set up phone calls.
- **TPCC**: Third Party Call Control enables an application to control the telephone network to set up calls automatically.
- **OWLCS**: Oracle WebLogic Communication Services. Offers the TPCC service to Fusion applications and sets up the calls via SIP integration with the telephony network.
The set up task list Define WebLogic Communication Services Configuration delineates five tasks required for the correct configuration and implementation of Click to Dial. There is an optional task, separate from the set up task list, required for implementing Interaction logging.

Information about implementing Click to Dial can be found in the Oracle Fusion Applications Administrator's Guide. Detailed information about configuring and maintaining WebLogic Communication Services is found in the Oracle WebLogic Communication Services Administrator's Guide.

**Configure and Deploy WebLogic Server**

- Deploy WebLogic Communication Services: After the Oracle WebLogic communication server is deployed, this manual task activates the server.

  See: Oracle WebLogic Communication Services Administrator's Guide

**Integrate Communications Services**

- Integrate WebLogic Communication Services with Telephony Network: This manual task integrates communications within the telephony environment. OWLCS must be configured to interface with the specific characteristics of the telephony network.

  See: Managing Oracle WebLogic Communication Services for Click to Dial Functionality

**Specify the Domain and Address**

- Manage Telephony Session Initiation Protocol Domain: The address of the SIP PSTN gateway or SIP softswitch serving the users within that domain. This address is needed by Click to Dial to correctly form the SIP addresses required by WebLogic Communication Services. For example: SIP:+1650-555-1212@pstn_gateway.oracle.com;user=phone where pstn_gateway.oracle.com is the SIP domain. The SIP domain can also be expressed in the format 10.1.1.1 or 10.1.1.1:5060 (IP address or IP address with port number). Profile Option: Define SIP Call Domain.

- Manage Third Party Call Control Web Service Host and Port: Required by Click to Dial to send the call setup commands to the WebLogic Communication Services TPCC service. Specify the host name and port for the WebLogic Third Party Call Control web service. Profile Option: Specify TCPP Host Port.

**Enable Click to Dial**

- After configuring the server and defining the SIP domain and TPCC, perform the Enable Click to Dial task. This task sets the value of the profile option Enable Click to Dial to 'Yes.'

**Call Logging via Fusion Interactions**

- To initiate the Interaction based logging for Click to Dial, set the profile option Call Interaction Logging Enabled to 'YES.'
Define Common CRM Configuration

Define Source Systems

Source Systems: Explained

Source systems are used to import data into Oracle Fusion Applications, and are used within the application to identify source data information. You can specify whether the source system is a Spoke system, such as a legacy system, or a Purchased system, such as data from a third party provider. You can also specify what type of data will be imported using the source system, for example, you can specify that a source system will import trading community members.

You can configure the following for a source system:

- Source system code, name, and description
- Source system type
- Enable for Items, Trading Community Members, and Order Orchestration and Planning

Source System Code, Name, and Description

You can create a source system code to uniquely identify the source system. Source system codes are used by the application when creating references between source IDs and the Oracle Fusion Applications database IDs. You can create a source system name and description to provide information that is more descriptive than the source system code.

Note

You cannot update the source system code once you have created the source system.

Source System Type

You must set up a source system as either a Spoke system, such as a legacy system, or a Purchased system, such as data from Dun & Bradstreet.
Enable for Items, Trading Community Members, and Order Orchestration and Planning

You should select which types of entities will be imported from the source system into the Oracle Fusion Applications database from the following:

- Items
- Trading Community Members
- Order Orchestration and Planning

You can select one or more of these entity types as required for the source system. It is important to enable the correct entity types because each import UI filters source systems based on their entity type. For example, if a source system is enabled for Trading Community Members and Items, then the source system can be selected as a data source in the Trading Community Members and Items import UIs, but the source system cannot be selected in the Orchestration and Planning import UI.

Source System Entities: Explained

Source System Entities are the entities, such as addresses and parties, which can be imported using a specified source system.

When you import data from a source system, all the entities in the source system data will be imported. Within the Source System Entities UI you can then choose to allow multiple source references, which allows mappings between one Oracle Fusion Applications database ID and multiple source IDs from the same source system.

FAQs for Source Systems

What happens if I allow multiple source system references?

Allowing multiple source system references means that when you import data from a source system you can merge multiple, or duplicate, source system records and create one record in the Oracle Fusion Applications database.

If you do not allow multiple source system references then an Oracle Fusion Applications database record will be created for every source system record. This means that you could potentially create duplicate records in the Oracle Fusion Applications database.

Define Identifiers

Identifier Types: Explained

You can set up additional identifier types to provide extensions to organization, person, or group party attributes. For example, you can create an additional identifier type to record a person’s passport number.
You can choose which party types can use the additional identifier type, and these can be either a person, organization, or group party type, or a combination of these party types. You can also choose whether the value of the identifier type has to be unique, for example you can specify that each entry of a passport number has to be unique by selecting **Yes** for **Unique Within Type**.

Additional identifier types do not automatically appear in the user interface. If you want to use identifier types in the application you will need to call the web service **Trading Community Member Name and Identifier Setup**

### Define Classifications

#### Classifications: Explained

The classifications model provides you with a flexible tool to categorize entities such as parties, projects, tasks, and orders. Classifications enable you to classify an entity, such as a party, in a way that the rest of the world sees it, in addition to the way that it is referenced within your organization.

The major components of classifications are:

- Classification categories
- Classification rules
- Classification codes
- Classification code hierarchy
- Entity assignment

#### Classification Categories

Classification categories give you the ability to classify entities under a broad subject area. For example, you can classify organizations based on the industries they operate in. Classification categories are a logical grouping of one or more classification codes and allow classification code rules to be defined.

#### Classification Rules

Classification categories can have rules that define how classifications can be assigned to entities. When you set up classification categories specific rules can be created, such as allowing the parent classification code to be assigned to a party, and allowing multiple classification codes to be assigned to an entity.

#### Classification Codes

The individual values within the classification category are called classification codes. For example, in the 1987 SIC classification category there is a classification code of software that can be assigned to a party in the software industry. You can organize classification codes into a hierarchical tree, with a parent classification code at the top of the tree and child classification codes branching off from the parent code or other classification codes.
Classification Code Hierarchy

You can create hierarchies of classification codes within a classification category. For example, you can set up a classification category of IT containing the classification codes hardware, keyboards, and printers. You can then set up the classification code of hardware as the parent code at the top of the tree, with the classification codes of keyboards and printers as child codes underneath. You can create further child classification codes, such as dot matrix, ink-jet, and laser below the printer classification code.

Entity Assignment

Define which entities can be assigned to a classification category by entering the entity table name and creating a Where clause in SQL. Only entities that satisfy the Where clause are assigned the classification category. For example, a classification category called industries with the Where clause of where "party_type = ORGANIZATION" would have the result that only organizations can be classified with the industries classification category.

FAQs for Define Classifications

What happens if I allow parent code assignment?

You can assign the parent classification code to an object, as well as the child classification codes. The parent classification code is the code at the top of the classification code tree.

If you don’t allow parent classification codes to be assigned to an object, then you can assign only child classification codes, or codes that are below another classification code in the tree, to an object.

What happens if I allow multiple class code assignment?

You can assign more than one classification code from this classification category to an object.

If you don’t allow multiple classification codes to be assigned to an object, then you can assign only one classification code from this classification category to an object.

Can I update an existing entity assignment rule?

No. You can delete the entity assignment rule and create a new one.

Define Resource Information

Defining Notes: Points to Consider

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:
• Note Types
• Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn't exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article Extending CRM Applications: how it works.

Resource Directory: Explained

The Resource Directory offers detailed information about all the resources within the deploying organization. The Resource Directory also enables you to find and communicate with other resources, and to network and collaborate with them.

Use the Resource Directory to perform the following tasks:

• View and modify your profile
• View your organization and team membership information
• View information related to other organizations and teams
• View the profiles of other resources
• Communicate with other resources

Setting up Resources: Explained

Setting up resources involves identifying a person as a resource and specifying optional profile details as needed. This is an important step because until you identify users as resources, you cannot assign work objects to them.
While identifying a resource is the only mandatory task in resource setup, you may also need to perform some of the following tasks while setting up resources.

- Specify the end date for a resource's engagement with the deploying company
- Assign roles to resources
- Assign resources to organizations
- Assign resources to teams

**Identifying Resources: Explained**

The Identify Resources step in the Manage Resources task is only needed to identify an existing employee, contingent worker, or partner member as a resource. Usually they are identified as resource in the Manage Users task, or in the Partner Center. If you have created partner members or internal users in the system without making them resources, you can identify them as resources in the Identify Resources step. Until you identify employees, contingent workers, and partner members as resources, you cannot assign them work objects.

**Note**

Resources need not necessarily belong to an organization, nor do they need to have specific roles assigned. However, it is best to always associate resources with an organization either as managers or as members. Similarly resources should also have at least one role as part of their organization membership. When you identify users as resources, all you indicate is that these new resources can now be assigned work within the deploying company.

**Resource Skills and Resource Assignment: Explained**

Resource skills help you assign resources to organizations and teams which can best utilize a specific set of skills. For example, if a resource is skilled in a specific technology, product, or business domain, you can assign the resource to teams and organizations that need resources possessing such skills. Use skill-based resource assignment to get the best out of the resources available to the deploying company.

**Resources: How They Work within a Team**

You can include resources from different resource organizations to work together on a work object as members of the same resource team. You can also include entire resource organizations into a resource team. Generally what resources can do is controlled by their resource organization membership and their hierarchy. Resource teams provide a flexible way of bringing resources together without any organizational or hierarchy-based restrictions.

**Assigning Resources to Teams**

You can assign identified resources to teams and assign them roles within the team. Each resource can have a specific role within a team. Thus, a resource may play different roles in different teams.
FAQs for Manage Resources

What happens if I add a resource to the organization?

When you add a resource to an organization, the resource becomes a member of the organization. This positions the resource within the organization hierarchy.

Organization membership information is part of the publicly visible details of a resource profile. This means that a resource's organization membership and reporting structure are visible to all active resources within the organization.

If you assign the entire organization to a resource team, all member resources are automatically assigned to the team. This information also becomes part of the resource's publicly visible profile.

What's the difference between a partner resource and an internal resource?

The main difference between an internal resource and a partner resource is the company for whom each works. While the internal resource is an employee or contingent worker of the deploying company, the partner resource is an employee of the partner company.

The methodology used to create resources of these two types is also different. While the partner administrator or channel manager creates a new partner resource through the Oracle Fusion Partner Management applications, internal resources are added using the Manage Users, Hire Employee, or Import Person and Organization task.

Another difference between partner resources and internal resources is that partner resources cannot access the Resource Directory while internal resources can.

Can I create an employee or contingent worker resource?

No. You can only identify existing employees and contingent workers as resources in the Manage Resources task, but you cannot create a new employee or contingent worker in the Manage Resources task.

You can create an employee or contingent worker using Manage Users task, Hire Employee task, or Import Person and Organization task.

Define Resource Organization Information

Resource Organizations and Organization Usage: Explained

You can assign organization usage information to resource organizations to classify them based on how they can be used. For instance, resource
organizations engaged in sales activities can be assigned the Sales Organization usage. This enables you to sort organizations based on their usage, simplifying your task of working with them.

**Resource Organization Hierarchy Versioning: Explained**

As organizations evolve, you may need to make changes to the existing organization hierarchy. Create organization hierarchies to capture these changes without impacting active hierarchies.

Depending on the urgency and nature of the changes within the deploying company, organization hierarchy changes can either be immediate or planned.

**Implementing Immediate Resource Organization Hierarchy Changes**

In case of immediate changes in the organization hierarchy, either make changes directly to the hierarchy or create a new version of the existing hierarchy and set it to become active when the new organization structure needs takes effect.

---

**Note**

Changes made to existing hierarchies are saved automatically and updated immediately.

---

**Implementing Planned Resource Organization Hierarchy Changes**

Create a new version of the active hierarchy and specify the date on which the new version needs to become active. Once the new version is saved, you can make and save the changes needed. Ensure that you have made all the changes needed to the new version before the date on which the new version needs to become active.

**Define Resource Role Information**

**Define Resource Roles: Explained**

Defining resource roles involves defining and configuring the roles that a resource plays as an individual or within a resource organization or resource team in the deploying company. This requires you to specify who a resource is within the enterprise and what specific role the resource performs within the context of an organization or team.

You can assign defined roles to resources directly or to resources within an organization or team context. This action simplifies the task of individually assigning complex roles to resources within the organization.

You can also set several flags while defining roles. Use these flags along with the organization hierarchy information to define the reporting hierarchy of resources. Use the Manager flag to tag a role as a supervisor role. Similarly,
attach a Member tag to a role to make it a subordinate role in the hierarchy. Tag roles as Administrator or Lead to indicate the roles that the resource roles have within the hierarchy. Additionally, you can use these flags along with the organization hierarchy information to maintain manager-to-manager relationships within the organization.

**Assigning Resource Role Types**

Resource role types organize roles into logical groups. This simplifies role assignment and assignment tracking. For example, the Partner resource role type defines a set of partner-specific roles such as partner sales representative and partner sales manager. Use the Partner resource role type to determine the roles that are appropriate for partner members. Similarly, use the Sales resource role type and the Marketing resource role type to categorize the appropriate sales and marketing roles for internal employees or contingent worker resources.

**Provisioning Security Roles: Explained**

Security role provisioning is the process of automating the provisioning and de-provisioning of security roles based on resource role assignment to resources. Once security roles are provisioned to resources, they can access the tasks and data enabled for the security role.

**Difference between Resource Roles and Security Roles**

Resource roles indicate who a person is to the deploying company. As such, resource roles are used for filtering resources and for generating reporting hierarchies in addition to being used to define security policies. A key difference between a security role and resource role is that a resource role may be assigned to a resource without a user account, while a security role can only be provisioned to a resource who has a user account. So while in some cases the resource role may be defined at the same granularity as the security role and used to automate security role provisioning, the resource role concept remains separate from security roles.

**Job-to-Role Mapping: Explained**

In the Manage Resource Roles task, you can establish job mapping for a resource role. Job-to-resource-role mapping enables you to associate HCM jobs with specific resource roles. This mapping simplifies the task of assigning resource roles to new employees or contingent workers, resulting in time and costs efficiency.

For example, suppose a new employee joins the IT department as a data quality manager. If the new employee's job is already mapped to a resource role like Data Steward Manager, the resource role is automatically assigned when the employee is identified as a resource in the system. This enables you to place new employees faster in organizational and reporting hierarchies. If security roles are also associated with the resource role, then the new employee's access privileges are also granted automatically.
FAQs for Define Resource Team Information

What's a resource team?

A resource team is a group of resources formed to work on work objects. A resource team may comprise resource organizations, resources, or both. A resource team cannot be hierarchically structured and is not intended to implement an organization structure. You can also use resource teams as a quick reference to groups of related resources that you can quickly assign work objects to.

Note

Members of teams can either be reassigned separately, or entire teams can be assigned to other tasks as required.

Define Products: Define Basic Items

Managing Product Value Sets: Explained

Value sets are specific to the application in which they will be used. In the Oracle Product Information Management application, value sets are used primarily for defining attributes where the values that an attribute can have is limited to a specific set of values.

Value sets can be edited or new value sets can be created from the Manage Product Value Sets page. The Edit icon launches the Edit Value Sets page, which redraws in the same region of the local area. The Create icon launches the Create Value Sets page, which redraws in the same region of the local area.

A value set is defined by the value set code and is specific to the module of an application in which the value set is to be used, such as Item Class.

The validation type determines how the value of field is validated for the assigned value set. The following are the seeded values:

- Format Only
- Independent
- Dependent
- Subset
- Table

The value data type determines the data type for the value set. The following are the seeded values:

- Character
- Number
• Date
• Date/Time

Managing Product Child Value Sets: Explained

The Manage Product Child Value Sets task uses the same page as the Manage Product Value Set task.

A child value set is used to define variants for stock-keeping units or SKUs. A SKU contains the common properties for an item. For example, a shirt can be produced with colors; white, red, yellow, and blue. The variant is used to represent the colors of the shirt.

You define child value sets as follows:

• Create a value set with validation type of independent, for example All Colors.

• Select the new value set in the Manage Product Value Sets results table, for example All Colors.

• Click Manage Values, create several values, for example Blue, Red, Green, Yellow, and Black.

• Create a value set with validation type of Subset and enter the first value set you created for the independent value set, for example: Summer Colors.

• Select the value set Summer Colors in the Manage Product Value Set result table.

• Click Manage Values and then click the Add icon. The dialog will show a list of values based on the value set named Summer colors. Select two of them.

The value set Summer Colors is a child of All Colors.

Managing Default Item Class: Explained

The Root Item Class is seeded and all item classes are created as children of the Root Item Class. For Oracle Fusion Product Model customers, only the Root Item Class is available. The Manage Default Item Class task enables Product Model customers to manage item class templates, descriptive flexfields, attachment categories and lifecycle phases. The Manage Default Item Class task launches an edit page for the Root Item Class.

The functionality for the Root Item Class is defined using three tabs:

• The Basic tab enables descriptive flexfields and attachment categories to be viewed and managed for the Root Item Class.

• The Lifecycle Phases tab enables one or more lifecycle phases of a lifecycle to be associated with an item class.
• The Templates tab is where you define and manage item templates for the item class.

Managing Item Statuses: Explained

In the Item Status table, select a status code to display the associated attribute groups and attributes as well as control information.

You can create or edit or delete item statuses on the Manage Item Statuses page. Inactive dates are used to specify the date after which the item status will no longer be active. Operational attribute groups and attributes corresponding to the selected item status are displayed in the Details section. Select a value for the status from choice list for the attribute. Whenever the status is applied to the item, the value of the attribute may change. If the status will have no value, select No.

Select the Usage value of None or Defaulted or Inherited in the choice list for the Usage field that corresponds to how the attribute value will change based on the item status value:

• Defaulted-Sets the values of the item status attributes when the status value changes, but allows the overriding of the value during import and update of item.

• Inherited-Sets the values of the item status attributes when the status value changes, but overrides cannot occur.

• None-The item status attribute values will not be changed.

Any change made to an item status is not applied automatically to existing items, but will be applied during the editing of an item when the item status value is changed.

Managing Item Types: Explained

Item types are managed on the Manage Item Types page.

There are 32 seeded item types and you can edit them or create additional item types.

Item types are date-enabled and are made active or inactive by adjusting the Start Date and End Date.

To benefit from the use of item types, you must enable them by selecting the Enable checkbox.

Managing Cross-Reference Types: Explained

Cross-References provide the functionality to map additional information about an item in the form of a value and cross-reference type. For example, the cross-reference can map between an item and an old part number, where the value is the value for the old part number and the type is Old Part Number. Cross-
Reference Types are part of item relationships where the item relationship type is Cross-Reference. There are no values seeded for cross-reference types. You define the values using the Manage Cross Reference Types task. Cross-reference types are date-enabled and can be made active or inactive by adjusting the values of the Start Date and End Date. To benefit from using item relationship for cross-reference, you must enable cross-reference types by checking the Enable checkbox.

Importing Items: Explained

You can import items and item-related information using interface tables. This import data is loaded into the production tables using the Import Item task in Functional Setup Manager.

**Import Item**

The Import Item task in Functional Setup Manager creates an Enterprise Storage Server (ESS) process that takes the data that is loaded in the interface tables and uses the import process to move the data to the production tables. The import processes will perform all of the validations necessary to ensure the data imported is correct prior to moving the data into the production tables.

Access the Enterprise Storage Server and provide a process name (job definition) such as Item Import Process.

In Functional Setup Manager, access the All Tasks tab on the Overview page, and search for the Import Item task with the name of your ESS process definition, then click the Go to Task icon in the search results for that Import Item task.

The parameters for the item import process are

- **Batch ID**: Associate the interface table to an item batch definition.

- **Organization**: Select an organization to be used for the import.

- **Process Only**: Determines how the data is processed. The choices are:
  - Create
  - Sync
  - Update

- **Process All Organizations**: Select Yes if the import contains items that will be imported to multiple organizations.

- **Delete Processed Rows**: Select Yes to delete rows that are imported without errors

Click Submit and the Request Number will be displayed.

**Monitoring Import Items**

Access Monitor Item Imports in Functional Setup Manager to search for specific Enterprise Storage Server processes and monitor their status in the search results table.
Managing Related Item Subtypes: Explained

A related item is an item relationship between two existing items. How the two items are related is defined by a subtype. Multiple subtypes for related items are seeded, and you define additional subtypes on the Manage Related Item Subtypes page.

Managing Descriptive Flexfields for Items: Explained

You can use descriptive flexfields to capture additional information about items beyond what is provided by the predefined set of operational attributes in Oracle Fusion Product Model.

Item Descriptive Flexfields

If you are not using Oracle Fusion Product and Catalog Management, then you cannot create user-defined attribute groups and attributes. However you can use descriptive flexfields associated at Item level to create fields to capture information about items. Like other descriptive flexfields, item descriptive flexfields have context segments and context-sensitive segments whose values are validated on entry by value sets. You can define the value sets to control what values users can enter in a descriptive flexfield segment. Examples of information that you might capture are size and volumetric weight.

Manage this flexfield type by using the Manage Item Descriptive Flexfields task, which you can access by searching for flexfield tasks on the Setup and Maintenance Overview page.

Item Revision Descriptive Flexfields

Use descriptive flexfields associated at Item Revision level to capture item revision information whose values may differ between revisions of the same item.

Manage this flexfield type by using the Manage Item Revision Descriptive Flexfields task, which you can access by searching for flexfield tasks on the Setup and Maintenance Overview page.

Item Relationship Descriptive Flexfields

When defining descriptive flexfields associated with item relationships, you must use certain prefixes when naming the context segments, in order for the segments to be displayed for the respective relationships.

The prefixes required for naming the context segments are listed in the following table, with their corresponding item relationship types. For example, if you define an item relationship descriptive flexfield with a context segment named RELATED_RELATIONSHIP_ATTRIBUTES, then the value segments of this context will be displayed for Related Item Relationships when users conduct transactions in that context. For another example, when users navigate to a UI of a particular object,
such as a Competitor Item, they see the contexts whose internal name has the prefix COMP.

<table>
<thead>
<tr>
<th>Relationship Type</th>
<th>Prefix for Context Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Competitor Item Relationship</td>
<td>COMP</td>
</tr>
<tr>
<td>Customer Item Relationship</td>
<td>CUST</td>
</tr>
<tr>
<td>Item Cross-reference Relationship</td>
<td>XREF</td>
</tr>
<tr>
<td>GTIN Relationship</td>
<td>GTIN</td>
</tr>
<tr>
<td>Manufacturer Part Number Relationship</td>
<td>MFG</td>
</tr>
<tr>
<td>Related Item Relationship</td>
<td>RELATED</td>
</tr>
<tr>
<td>Source System Item Relationship</td>
<td>SYS</td>
</tr>
</tbody>
</table>

Manage this flexfield type by using the Manage Item Relationship Descriptive Flexfields task, which you can access by searching for flexfield tasks on the Setup and Maintenance Overview page.

**Trading Partner Item Descriptive Flexfields**

When defining descriptive flexfields associated with trading partner items, you must use certain prefixes when naming the context segments, in order for the segments to be displayed for the respective trading partner type.

The prefixes required for naming the context segments are listed in the following table, with their corresponding trading partner types. For example, if you define a trading partner item descriptive flexfield with a context segment named COMP_TPI_ATTRIBUTES, then the value segments of this context will be displayed for Competitor Item when users conduct transactions in that context.

<table>
<thead>
<tr>
<th>Trading Partner Type</th>
<th>Prefix for Context Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Competitor Item</td>
<td>COMP</td>
</tr>
<tr>
<td>Customer Item</td>
<td>CUST</td>
</tr>
<tr>
<td>Manufacturer Item</td>
<td>MFG</td>
</tr>
</tbody>
</table>

Manage this flexfield type by using the Manage Trading Partner Item Descriptive Flexfields task, which you can access by searching for flexfield tasks on the Setup and Maintenance Overview page.

**FAQs for Define Basic Items**

**What's the difference between lifecycle phase types and lifecycle phases?**

Lifecycle phase types are seeded and describe the type of lifecycle phase. They are Design, Obsolete, Preproduction or Prototype, and Production.

Lifecycle phases must be created by the user by selecting one of the seeded lifecycle phase types.
Manage Units of Measure

Units of Measure, Unit of Measure Classes, and Base Units of Measure: How They Fit Together

Define units of measure, unit of measure classes, and base units of measure for tracking, moving, storing, and counting items.

The Quantity unit of measure class contains the units of measure Box of 8, Box of 4, and Each. The unit of measure Each is assigned as the base unit of measure.

Unit of Measure Classes

Unit of measure classes represent groups of units of measure with similar characteristics such as area, weight, or volume.

Units of Measure

Units of measure are used by a variety of functions and transactions to express the quantity of items. Each unit of measure you define must belong to a unit of measure class.

Base Units of Measure

Each unit of measure class has a base unit of measure. The base unit of measure is used to perform conversions between units of measure in the class. For this
reason, the base unit of measure should be representative of the other units of measure in the class, and should generally be one of the smaller units. For example, you could use CU (cubic feet) as the base unit of measure for a unit of measure class called Volume.

Assigning Base Units of Measure to Unit of Measure Classes: Examples

Each unit of measure class must have a base unit of measure.

Scenario

This table lists examples of unit of measure classes, the units of measure in each unit of measure class, and the unit of measure assigned as the base unit of measure for each unit of measure class. Note that each base unit of measure is the smallest unit of measure in its unit of measure class.

<table>
<thead>
<tr>
<th>Unit of Measure Class</th>
<th>Units of Measure</th>
<th>Base Unit of Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>dozen, box, each</td>
<td>each</td>
</tr>
<tr>
<td>Weight</td>
<td>pound, kilogram, gram</td>
<td>gram</td>
</tr>
<tr>
<td>Time</td>
<td>hour, minute, second</td>
<td>second</td>
</tr>
<tr>
<td>Volume</td>
<td>cubic feet, cubic centimeters, cubic inches</td>
<td>cubic inches</td>
</tr>
</tbody>
</table>

Defining Unit of Measure Standard Conversions: Examples

A unit of measure standard conversion specifies the conversion factor by which the unit of measure is equivalent to the base unit of measure.

Scenario

This table lists examples of unit of measure classes, one unit of measure included in each class, the base unit of measure for the unit of measure class, and the conversion factor defined for the unit of measure.

<table>
<thead>
<tr>
<th>Unit of Measure Class</th>
<th>Unit of Measure</th>
<th>Base Unit of Measure</th>
<th>Conversion Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>dozen</td>
<td>each</td>
<td>12 units</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1 dozen = 12 each)</td>
</tr>
</tbody>
</table>
FAQs for Units of Measure

What's a unit of measure standard conversion?

A unit of measure standard conversion defines the conversion factor by which the unit of measure is equivalent to the base unit of measure that you defined for the unit of measure class. Defining a unit of measure standard conversion allows you to perform transactions in units other than the primary unit of measure of the item being transacted. The standard unit of measure conversion is used for an item if an item-specific unit of measure conversion has not been defined.

What's a UOM interclass conversion?

A UOM interclass conversion defines the conversion between the source base unit of measure ("From Base UOM") in one unit of measure class ("From Class") and the destination base unit of measure ("To Base UOM") in a different unit of measure class ("To Class").

For example, the item is gasoline. The From Base UOM (of the From Class called "volume") is liters. The To Base UOM (of the To Class called "quantity") is Barrels. The conversion is 158.76 liters (volume) to 1 barrel of oil (quantity).

What's a UOM intraclass conversion?

A UOM intraclass conversion specifies the conversion between a unit of measure (the "From UOM") and the base unit of measure of the same class.

For example, the item is soda pop. The unit of measure class is Quantity. The From UOM is Case (CS). The base unit of measure is Each (EA). The conversion is 24, to specify that 1 CS = 24 EA.

Define Products: Define Advanced Items

Transactional Attributes: Explained

Attributes that exist for each instance of an item and the values for the attributes can be different for different instances of an item.

For example:

- The number of megabytes (MB) or gigabytes (GB) of e-mail storage on a digital subscriber line (DSL) account.
• The monogram text on a shirt pocket.
• The color of a music player.

These attributes are defined at the item class and their attribute value is captured at the time of a transaction by downstream applications. The metadata values of these attributes are maintained at the item class. Order orchestration and order capture are two examples of applications currently using transactional item attributes. All transactional attributes must be associated with a value set.

The following metadata values can be defined for an attribute.

• Required: Indicates whether the attribute value is required at the transaction.
• Default Value: Indicates the default value of the attribute.
• Value Set: Indicates the value set associated with the attribute.
• Read Only: Indicates whether the attribute value is read only.
• Hidden: Indicates whether the attribute is not shown.
• Active: Indicates whether the attribute is active or inactive.

Transactional attributes are inherited across the item class hierarchy. The metadata is data-effective. Changes in the metadata will be reflected immediately at the item level. For example:

• Any of the metadata of a TIA belonging to a specific domain, if modified in the child item class would break the inheritance. Any changes done at the parent item class for this TIA would not get inherited. Multiple records with same date range can exist if they belong to different domains. For example, TIA Memory is associated with Domain DOO and Order Capture. Each of the domains may use a different set of metadata for its own purpose. Hence, for the same date range, two different records can exist. Only Start Dates for a TIA would be entered by a user. End date would be calculated automatically based on the next Date Effective record.

• Users can modify (either Start Date and metadata) of a future effective record. Records with Starting date as Past cannot be modify or edited.

• Only start dates can be set to permit updating by a user, and the end date of a record will automatically be pulled from the next record.

• Any changes performed in the parent item class would be inherited by the child item class. If the corresponding record is modified in the child, then these changes will not be inherited.

Item pages provide a mechanism with which to customize the user interface.

**Pages and Attribute Groups**

Pages and attribute groups enable you to structure your data.

Benefits include:

• You can combine and sequence attribute groups into pages.
- There is no limit on the number of attribute groups associated with a page

- Pages can be created at item class and are inherited down the item class hierarchy.

- Attribute groups can be added to pages sequentially and based on this sequence, these attribute groups are shown in items.

- Attributes groups can be added for an inherited page at the child item class.

Functional Item pages are another type of special pages which are used to associate pages already created for use in the application. Application scope indicates the application which uses these pages and the usage indicates the specific use of the configured pages.

**Data Quality**

You can associate attributes for the purpose of standardization and matching, to be performed when items are created. You restrict the attributes to be processed for standardization or matching or both. Selecting Standardization allows the data quality engine to return the standardized values for these attributes. Matching allows the data quality engine to return any existing items which matches the value of these attributes and are potential duplicates.

**Lifecycle Phases**

Sequential lifecycles phases enable you to track and control the lifecycle phases of items. Each phase represents a set of tasks and deliverables that are required before promoting the item to the next phase. You can associate lifecycle phases to an item class which are created elsewhere. Lifecycle phases are inherited down the item class hierarchy and new lifecycle phases can be added to child item classes. For example, the lifecycle phases for a computer component item class might be: Concept, Prototype, Production, and Retirement.

**Templates**

Template is a defined set of attribute values used during item creation. When you apply a template to an item, you overlay or default-in the set of attribute values to the item definition. For example, every time users in a particular organization create new items, the attributes, as defined and approved by the organization appear in the appropriate fields. No user guesswork is required, and time is saved during the creation of items with a similar form, fit and function. Templates are created for each item class. Templates are specific to organization. Templates are inherited down the item class hierarchy. You can define both operational attributes and user defined attributes for each template.

**Search and Display Format**

Search formats provide a convenient way to save frequently used search criteria. Search formats created at item class will be available to all users. Search formats are always created in the context of item class. Display formats enable you to predefine search display views. You can use these views to look at different sets of item attributes that are returned by the search. Display formats created at item class will be available to all users. Display formats are always created in the context of item class.
Import Format

An import format identifies the base and user-defined attributes in an item class that are imported into the application using a spreadsheet. Consequently, when you import item business entities from a spreadsheet, the items are all imported into the particular item class defined in the import format. These imported item business entities inherit all the attribute groups defined for the specific item class. You cannot edit the layout of an import format once it is created.

Managing Item Classes: Explained

When you are ready to create or edit an item class, you must decide whether to allow items to be created under the item class.

To create an item class, perform the following steps:

- Create a list of all items.
- Classify or categorize these items (Item classes).
- Define any parent child relationships (Item Class Hierarchy).
- Gather the unique types of specifications required for each type of classification at a high level (user-defined attribute groups).
- Gather the unique specifications required within the group (user-defined attributes).
- If there are specified values that must be used, define them (value sets and values).

Item classes can be used for classification purposes and in some case, item creation may not be allowed. By optionally setting the item creation allowed attribute to No, item creation under an item class can be prevented. However, a child item class of such a item class may be allowed for item creation. For example:

This will prevent items from being created in Computers and Desktop and allow items to be created for Green Desktops and Gaming Desktops. Optionally, specify a date on which the item class will become inactive. You cannot specify an inactive date that is later than the inactive date of an item class parent, nor can you specify an inactive date that has already passed. Also, all children of a parent item class with an inactive date should be made inactive at the same time or earlier.
Attachment categories enable you to categorize and classify attachments to an item. To classify item attachments, associate attachment categories with item catalog categories. Associated attachment categories are inherited down through the item class hierarchy.

Item classes can be set up for item numbers and descriptions so that they are automatically generated when net items are created. This ensures that new items created in item classes have a consistent numbering scheme. Versioning control allows new versions to be created for all items of item class. An integrated workflow definition allows creation of custom new item request definition for new items created. This definition workflow enables you to route the definition and approval of an item using various steps. When creating a new item, various aspects of an item like base operational attributes, user-defined attributes, structures, attachments, categories associated, organization assignments, are defined by various people in the organization using a workflow process.

You can control item creation, viewing and update access by assigning a role on the item class to a principal or group of users. Security allows a person or a group to have privileges on an item of item class in each organization. This is inherited and hence a person who has a privilege in a parent item class will automatically have the same privilege in the child item classes.

**FAQs for Define Advanced Items**

**What are item classes?**

The item class hierarchy provides a logical classification and grouping of similar products, and also acts as a template for product definition by enabling the association and inheritance of data elements and policies that are shared by products.

**How can I create an item class?**

To create an item class, select a parent item class on the Item Class Search Results page and select Create. Provide the required information, and optionally include additional details, such as attribute groups, pages, templates, and search and display formats.

**Define Sales Catalogs**

**Functions and Miscellaneous Actions: Explained**

You can set various options to customize the runtime instance of your product catalog.

The Functions and Miscellaneous tabs have several built-in features and options for you to choose from.

**Functions**

Select certain functions and specify how they should run depending on your processes.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Engine</td>
<td>Determines the availability of a product in stock.</td>
<td>Do not run</td>
<td>Do not call the availability service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quick availability</td>
<td>Show whether the product is available or out of stock</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Detail availability</td>
<td>Show the number of quantity available. Example in stock.</td>
</tr>
<tr>
<td>Eligibility Engine</td>
<td>Determines the eligibility of a product or product group for a customer or a geographical area.</td>
<td>Do not run</td>
<td>Do not call the eligibility service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Run and hide</td>
<td>Call the eligibility service but hide the ineligible products, product groups and promotions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Run and show</td>
<td>Call the eligibility service and show the ineligible products, product groups and promotions with the appropriate message.</td>
</tr>
<tr>
<td>Pricing Engine</td>
<td>Determines the price for a product.</td>
<td>Do not run</td>
<td>Do not call the pricing service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Complex</td>
<td>Show the List Price, Your Price, Discount, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Simple</td>
<td>Show the List Price only.</td>
</tr>
<tr>
<td>Territory Engine</td>
<td>Determines the products in a territory.</td>
<td>Do not run</td>
<td>Do not check for territory information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Enforce territory</td>
<td>Always show the products and product groups that are part of the territory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Display user choices</td>
<td>Provides the user with the choice to toggle between all product or product groups and the ones that belongs to the territory only.</td>
</tr>
</tbody>
</table>

**Miscellaneous Actions**

Set preferences such as button label, sort by text, number of products per page, and so on.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add item button label</td>
<td>The selected value is shown next to the product in the runtime interface.</td>
</tr>
<tr>
<td>Add category button label</td>
<td>The selected value is shown next to the catalog or category in the runtime interface.</td>
</tr>
<tr>
<td>Add category enabled flag</td>
<td>Allows buttons to be shown next to the catalog or categories.</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>Records per page</td>
<td>Number of records to be displayed per page.</td>
</tr>
<tr>
<td>Sort by format text</td>
<td>Sort format of the entire label that you want displayed in the runtime interface. The default pattern is [ATTR]: [SORT_ORDER]. Example: Name: A to Z.</td>
</tr>
<tr>
<td>Sort by product label prefix</td>
<td>Sort format of the prefix label that you want displayed. Example: If the default is Name: A to Z, you can select an alternate label for Name. It could be Item: A to Z.</td>
</tr>
<tr>
<td>Sort by sequence product ascending label</td>
<td>Sort format of the ascending suffix label that you want displayed. Example: If the default is Name: A to Z, you can select an alternate for A to Z. It could be Name: Ascending.</td>
</tr>
<tr>
<td>Sort by sequence product descending label</td>
<td>Sort format of the descending suffix label that you want displayed. Example: If the default is Name: Z to A, you can select an alternate for Z to A. It could be Name: Descending.</td>
</tr>
<tr>
<td>Sort by sequence ascending first flag</td>
<td>Select Yes to display ascending labels first in the Sort By LOV.</td>
</tr>
<tr>
<td>Show immediate child products only</td>
<td>Shows immediate products of a given category disregarding the standard behavior of showing all products (including child categories) if narrow by is defined on the category.</td>
</tr>
<tr>
<td>Image server</td>
<td>Identifies the source of images for products and product groups.</td>
</tr>
<tr>
<td>Image server alternate path</td>
<td>Identifies an alternate image source location (URL)</td>
</tr>
<tr>
<td>Enable transactional attribute</td>
<td>Allows transactional attributes to show up in product detail page. Transactional attributes are attributes that can be selected such as color and size of shirt.</td>
</tr>
<tr>
<td>Hidden category optional attribute list</td>
<td>You can specify the attributes you would like to hide from the category UI here. This could be a comma separated list of attributes that needs to be hidden from the category list page.</td>
</tr>
<tr>
<td>Hidden product optional attribute list</td>
<td>You can specify the attributes you would like to hide from the product UI here. This could be a comma separated list of attributes that needs to be hidden from the product pages.</td>
</tr>
<tr>
<td>Hide quantity</td>
<td>Set this to Yes, to hide the quantity field shown in the product page.</td>
</tr>
<tr>
<td>Hide unit of measure</td>
<td>Set this to Yes, to hide the unit of measure field shown in the product detail page.</td>
</tr>
</tbody>
</table>

**Change Display Options for Product Group Categories: Worked Example**

Megan and the marketing team notice that when they browse the ComfyGooseCatalog, availability information is shown against each product.
Since they have no visibility around the Loveseats product, the marketing team suggests to hide the availability information for the Loveseats category alone. Megan remembers from the training that Fusion Sales Catalog provides the ability to override the default behavior.

Megan recalls that she has to make this change from the Display Options for the Loveseats catalog.

**Change the Display Options**

1. Select the catalog from the Product Group Administration page and lock it.

   Megan chooses the Loveseats catalog.

2. Create a new entry in the Display Options tab.

   Megan creates a record and names it Hide Availability.

3. From the Applies To subtab, select the usage that this display option must be applicable to.

   Megan selects the Base usage.

4. From the Functions tab, select the Availability Engine Code function and assign Do Not Run as its value.

5. Save and publish the ComfyGooseCatalog.

   Megan validates the effect of this change at runtime.

**Display Options: Explained**

Use the display options to control various aspects of the published product group.

**Modifying Product Group Information**

You can make small but significant changes to a product group from these tabs. The changes here override default settings.

- **Applies To**

  Select the usage that this product group is applicable to. Usage defines the department or function within your organization for which this catalog is created.

  Apart from usage, you can also select the mode within the usage. Mode defines the department or function within your organization that uses the same catalog but with minor changes from other consumers.

- **Narrow By**

  Select Narrow By attributes and their appearance. These attributes appear as filters to narrow searches at runtime.

- **Template**
Select templates for category, product list, and so on.

- **Functions**
  Define the changes to specific settings in certain functions such as pricing and eligibility.

- **Miscellaneous**
  Change basic settings for the product group such as button label, number of items per page, invocation of the configurator, and so on.

## Creating a Sales Catalog: Worked Example

This example demonstrates how to create a sales catalog. In this example, ComfyGoose Inc is a state of the art outfit in the business of selling chairs and sofas. As part of their expansion plans, they recently bought Oracle Fusion CRM and are uptaking the best business practices and all the functionality it brings.

Megan works in the product marketing department and is excited about the sales catalog and the ease with which she can create products and catalogs. She gathers information about the categories to be created and the products that need to be associated to each category. She is familiar with the layouts and the navigation paradigm in the application.

As a first step, Megan decides to create a sales catalog.

### Creating a Sales Catalog

1. Log in to the application and select the Product Group Administration tab.
2. Click **Create** from the Product Groups pane.
3. Enter data in the relevant fields.

Megan enters the following details for her catalog.

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>ComfyGooseCatalog</td>
</tr>
<tr>
<td>Display</td>
<td>Comfy Goose Catalog</td>
</tr>
<tr>
<td>Description</td>
<td>Contains ergonomic chairs for your home or office needs, chairs for businesses such as call centers and offices; at attractive prices.</td>
</tr>
<tr>
<td>Root Catalog</td>
<td>Select to make this a root catalog. Only root catalogs can be added to a usage in Product Group Usage Administration.</td>
</tr>
</tbody>
</table>

4. Associate an image that should go with the catalog from the Details tab.
5. Add categories to this product catalog from the Subgroups tab.
Megan creates the following subgroups for the ComfyGoose Catalog product group.

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subgroup Name</td>
<td>• Chairs</td>
</tr>
<tr>
<td></td>
<td>• Sofas</td>
</tr>
<tr>
<td></td>
<td>• Medical Chairs</td>
</tr>
<tr>
<td></td>
<td>• Sports Chairs</td>
</tr>
<tr>
<td></td>
<td>• Chairs and Stools</td>
</tr>
</tbody>
</table>

6. Add further categories to a subgroup, if required. To do so, change the view of the product group from List to Tree view from the Product Groups pane. Select the category within which you want a subgroup. Megan creates further categories within some of the subgroups.

<table>
<thead>
<tr>
<th>Parent Subgroup</th>
<th>Sample Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chairs</td>
<td>• Heavy Duty/Call Center Chairs</td>
</tr>
<tr>
<td></td>
<td>• Ergonomic Chairs</td>
</tr>
<tr>
<td></td>
<td>• Leather Chairs</td>
</tr>
<tr>
<td></td>
<td>• Event Chairs</td>
</tr>
<tr>
<td>Sofas</td>
<td>Sofas and Loveseats</td>
</tr>
<tr>
<td>Sofas and Loveseats</td>
<td>• Leather Sofas</td>
</tr>
<tr>
<td></td>
<td>• Reception and Lounge Sofas</td>
</tr>
<tr>
<td></td>
<td>• Loveseats</td>
</tr>
</tbody>
</table>

7. Add products from the Products tab to the respective subgroups that you just created.

8. Click Publish to publish the product hierarchy that you just created.

A published catalog is available for use by different departments which is done via Usage Administration.

9. To associate a published catalog with a particular usage, click the Product Group Usage Administration link.

10. Select the usage, click the Product Groups subtab. Search and add the product group you just created.

Megan selects the Base usage and adds ComfyGooseCatalog to this usage.

Creating a Sales Catalog: Worked Example

This example demonstrates how to create a sales catalog. In this example, ComfyGoose Inc is a state of the art outfit in the business of selling chairs and
sofas. As part of their expansion plans, they recently bought Oracle Fusion CRM
and are uptaking the best business practices and all the functionality it brings.

Megan works in the product marketing department and is excited about the
sales catalog and the ease with which she can create products and catalogs. She
gathers information about the categories to be created and the products that
need to be associated to each category. She is familiar with the layouts and the
navigation paradigm in the application.

As a first step, Megan decides to create a sales catalog.

**Creating a Sales Catalog**

1. Log in to the application and select the Product Group Administration
tab.

2. Click **Create** from the Product Groups pane.

3. Enter data in the relevant fields.

   Megan enters the following details for her catalog.

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>ComfyGooseCatalog</td>
</tr>
<tr>
<td>Display</td>
<td>Comfy Goose Catalog</td>
</tr>
<tr>
<td>Description</td>
<td>Contains ergonomic chairs for your home or office needs, chairs for businesses such as call centers and offices; at attractive prices.</td>
</tr>
<tr>
<td>Root Catalog</td>
<td>Select to make this a root catalog. Only root catalogs can be added to a usage in Product Group Usage Administration.</td>
</tr>
</tbody>
</table>

4. Associate an image that should go with the catalog from the Details tab.

5. Add categories to this product catalog from the Subgroups tab.

   Megan creates the following subgroups for the ComfyGoose Catalog product group.

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Data</th>
</tr>
</thead>
</table>
   | Subgroup Name| • Chairs  
                 • Sofas  
                 • Medical Chairs  
                 • Sports Chairs  
                 • Chairs and Stools |

6. Add further categories to a subgroup, if required. To do so, change the view of the product group from List to Tree view from the Product Groups pane. Select the category within which you want a subgroup.
Megan creates further categories within some of the subgroups.

<table>
<thead>
<tr>
<th>Parent Subgroup</th>
<th>Sample Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chairs</td>
<td>• Heavy Duty/Call Center Chairs</td>
</tr>
<tr>
<td></td>
<td>• Ergonomic Chairs</td>
</tr>
<tr>
<td></td>
<td>• Leather Chairs</td>
</tr>
<tr>
<td></td>
<td>• Event Chairs</td>
</tr>
<tr>
<td>Sofas</td>
<td>Sofas and Loveseats</td>
</tr>
<tr>
<td>Sofas and Loveseats</td>
<td>• Leather Sofas</td>
</tr>
<tr>
<td></td>
<td>• Reception and Lounge Sofas</td>
</tr>
<tr>
<td></td>
<td>• Loveseats</td>
</tr>
</tbody>
</table>

7. Add products from the Products tab to the respective subgroups that you just created
8. Click **Publish** to publish the product hierarchy that you just created
   A published catalog is available for use by different departments which is done via Usage Administration
9. To associate a published catalog with a particular usage, click the **Product Group Usage Administration** link
10. Select the usage, click the Product Groups subtab. Search and add the product group you just created
    Megan selects the Base usage and adds ComfyGooseCatalog to this usage.

**Reusing a Sales Catalog: Worked Example**

This example demonstrates how you can reuse a sales catalog. In the scenario used in this example, ComfyGoose Inc is a state of the art outfit in the business of selling chairs and sofas. As part of their expansion plans, they recently bought Oracle Fusion CRM and are uptaking the best business practices and all the functionality it brings. Megan works in the product marketing department.

Megan gets a call from another division of the company, after a few months of deploying the ComfyGooseCatalog. They have heard of the new catalog that she helped build for the marketing team and enquire if they can have a similar subset for their division. They want to leverage as much as possible having a consistent look-and-feel.

An application developer must create an application and should be able to consume the sales catalog task flows to achieve this example. This task is similar to what the Sales module has done to consume the task flows provided by the sales catalog team.

Megan gathers the necessary information and identifies that this division primarily needs what is in the Chair category of the ComfyGooseCatalog. She is
very excited because she knows that she can simply reuse the Chairs category. In the past, she would have had to create another catalog repeating the same data, a maintenance overhead.

**Reusing an Existing Catalog for Another Department**

1. Navigate to the Product Group Usage Administration page
2. Create a new usage
   Megan creates the Call Center Division usage
3. Click the Product Groups subtab for the usage that you just created
   You can choose only product groups that are catalogs by themselves here. In other words, the catalog must be a root catalog.

Megan realizes that the Chairs catalog that she wants to reuse is a subgroup of the ComfyGooseCatalog. To make it a root catalog, she follows these steps:

   a. Navigate to the Product Group Administration page
   b. Select the product group Chairs from the Product Group pane
   c. From the Details tab, select Root Catalog
   d. Save and publish the catalog.
4. Add the product group to the new usage in the Product Group Usage Administration page.
   The Chairs catalog is now available for the Call Center Division.

**Enabling Filtering by Attributes: Worked Example**

This example demonstrates how to enable filtering by attributes. In the scenario used in this example, ComfyGoose Inc is a state of the art outfit in the business of selling chairs and sofas. As part of their expansion plans, they recently bought Oracle Fusion CRM and are uptaking the best business practices and all the functionality it brings. Megan works in the product marketing department. She has created the ComfyGooseCatalog and tried changing some usage attributes.

Megan now reviews the ComfyGooseCatalog with the team and decides that it will be nice to provide the user with few narrow by filters. After reviewing the definition of products associated to the Chairs category and its subcategories, it is decided that attributes Color and Material can be used for filtering.

The attributes must already be registered in the Product Group Attribute Administration page. These attributes must be present and associated in the Item Master.

**Associating Attributes for a Category**

1. From the Product Group Administration page, select the category to add the attributes for and lock it.
   Megan selects the Chairs category.
2. From the Filter Attributes tab, click **New** to select the attribute that you wish to associate to Chairs product group. Select the Advanced Search, Narrow By and Sort flags for the attribute. By selecting these flags, this attribute will get displayed in those regions.

**Note**

The attribute is already registered in the Product Group Attribute Administration page.

Megan selects the Color and Material attributes.

3. Create values for the selected attributes from the Attribute Values region below.

Megan creates Blue, Pink and Black as the values for the chair color. She also creates values for the chair material.

4. Megan saves the changes and publishes the Chairs product group.

Megan validated the changes in the graphical catalog. She can see the attributes that she created and their values in the Advanced Search, Narrow By and Sort options of the catalog.

### Using a New Template in a Catalog: Worked Example

This example demonstrates how to use a new template in a catalog. Megan informs the marketing team that the IT department has created a new template that gives a cool web 2.0 feeling and asks their permission to try it out. She informs them that it gives a carousel effect which is quite common while browsing songs, DVDs, and so on. The marketing team agrees to give it a try.

Using Fusion JDeveloper, application developers have to create a task flow using the Fusion ADF component and all necessary VOs to support the task flow. The task flow must be registered in the template administration.

Megan decides to use the template ComfyGoose Carousel template in the ComfyGoose Catalog’s browsing categories.

### Using a New Template in a Catalog

1. On the Product Administration page, Megan searches for the ComfyGooseCatalog and locks it.

2. In the Display Options tab for the selected catalog, create a new record.

   Megan creates a record and names it Carousel.

3. From the Applies To subtab, select the usage that this change should be applicable to.

   Megan selects the Base usage.

4. From the Templates subtab, create a record and select the appropriate template type and the new template.

   Megan selects the Category List Template as the type and the ComfyGoose Carousel template as the template.
5. Save your changes and publish the catalog.

Megan verifies the change in the catalog at runtime. The marketing team loves this cool effect and agrees to adopt it for the final catalog roll out.

Changing Usage Attributes for an Application: Worked Example

This example demonstrates how you can modify attributes for an application's usage. In this scenario, the Call Center division only deals with call centers across the world, they have not seen if any product they have sold is not applicable to call center companies. Therefore, they decide to switch off the eligibility and availability messages that show up in their side of the catalog.

Megan knows that this is a fairly simple task because she can do these settings from the Product Group Usage Administration page.

Modifying Attributes for an Application's Usage

1. Navigate to the Product Group Usage Administration page, and select the usage to make changes.

   Megan selects the Call Center usage.

2. From the Functions subtab, select the following:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Engine</td>
<td>Do not run</td>
</tr>
<tr>
<td>Eligibility Engine</td>
<td>Do not run</td>
</tr>
</tbody>
</table>

3. Save your changes and validate the effect in the catalog at runtime.

   Megan accesses the catalog and finds that the availability and eligibility messages do not appear anymore for the Call Center division's catalog.

Eligibility Rules for a Product Group: Explained

An eligibility rule can be a physical eligibility rule or a marketing eligibility rule

Some points to remember when you create an eligibility rule:

- An eligibility rule can determine one of the two kinds of eligibility for a product group, physical or marketing. They cannot be combined.

- A rule can be of two types: Available and Not Available. If contradictory rules are defined for the same product group, the Not Available rule will prevail.

- All children of a product group inherit rules from the parent product group.

- Eligibility rules set at the product level have precedence over the rules set at the product group level.
Physical Eligibility

When you create a physical eligibility rule, select attributes from these: Country, State or Province, City and Postal Code.

Marketing Eligibility

When you create a marketing eligibility rule, select a value for Customer Type.

FAQs for Define Sales Catalogs

How can I customize button labels in the catalog?

You can add custom labels for your catalog by specifying them in the Miscellaneous tab of Display Options. For example, you can change the default Add to Cart label and can select a different label such as Add to Shopping Cart. You can add an additional value for the button label from the Manage Product Group Lookups page from the Setup and Maintenance Overview page. You can enter a new value using the lookup Add Item Label Values. Once done, you can navigate to the product group administration to either override the button label for the entire catalog via the usage functions or to a specific product group from the display options tab.

Apart from customizing button labels, you can also set the number of items to display per page from the Miscellaneous tab.

What's the difference between rollup and sales catalog hierarchies?

A rollup catalog does not have the same product appearing multiple times within its hierarchy. The primary purpose of a rollup catalog is to create a hierarchy more tailored to forecasting purposes where a particular product appears only once in the entire hierarchy.

A sales catalog can have the same product appearing multiple times within its hierarchy. For example, the product Toys can be part of the Children category as well as the Electronics category within the same catalog.

Note

The Allow Duplicate flag distinguishes between a rollup catalog and a sales catalog. By default, the flag is selected making it a sales catalog. The Allow Duplicate flag is in the Details tab of the Product Group Administration page.

Why did some of the products in my published catalog disappear?

Products in your catalog are active for a specified period. Once the period expires, the product becomes inactive and does not appear as part of the published catalog. Activate the products from the Products tab of the Product Group Administration page.
What's a related group?

Related groups show the relationship between two different product groups. For example, a group that contains extended warranty products is related to a group that contains laptops. There are various relation types supported such as revenue, service and so on. This relationship is used in the other applications.

What's eligibility?

Eligibility is the condition of being qualified or being entitled for a purchase or enrollment.

There are two types of eligibility.

- Physical eligibility: Determines whether a service is available in a given area.
- Marketing eligibility: Determines whether a company is willing to sell a product, service, or promotion to a customer.

Some examples of eligibility are:

- Video Conferencing is available in California.
- Liquor is not available to minors.
- ESPN pay-per-view Requires HD Package

How can I change the labels for filter attributes at runtime?

In the Filter Attributes tab of the Product Group page, change the value for the Display of the attribute. The attribute can have an internal name but the name displayed at runtime can be more meaningful and customer friendly.

For example, if the attribute name is Laptop Color, you can change the display to Available Colors. This will display in the narrow by or advanced search options at runtime.

How can I use additional miscellaneous attributes?

An additional attribute is an extension to what is already provided in the application. For example, the currency attribute can be used if you wish to add an additional field to capture dollar values from the catalog and pass it to the consuming application. You may want to capture how much your customer is willing to pay for a product and send this information to an opportunity or a quote. You can expose this attribute with the value Customer Bid. In the runtime application, this field appears and the user enters an amount that she is willing to pay, say $90.

Why can't I modify a product group?

To modify a product group, you must first lock it. Click Lock in the Product Group page. If the product group is locked by another user, the Lock button is not visible and the Locked flag is enabled in the Details tab.
How can I associate promotions to a product group?

There may be several promotions running at any given time. You can associate active promotions to a product group from the Promotions tab. Coupons are a part of promotions and are added to the product group along with the promotion. You can also view the promotion's effective period.

What's the difference between the administration and published product group versions?

A product group can have two versions, Administration and Published. The Published version is visible to the end user in the graphical sales catalog. The administrator must make any changes to the product group in the administration version.

After publishing the administration version, the changes are made available to the published version and to the consuming applications.

How can I set exceptions for particular usages?

You can set exceptions for a usage from the Modes tab of the Product Group Usage Administration page. Exceptions can be made when you need minor modifications for different departments in your organizations, for the same usage. The following modes are available: Lead Management, Campaign Management, Opportunity Management, Territory Management, Opportunity Landscape, and Sales Prediction.

How can I know if my catalog is used by other catalogs?

You can get this information from the Product Groups Shared With region in the Product Groups pane. By default, all product groups are shared.

Define File-Based Data Import

Files, Import Objects, Mapping, and Import Activity Components: How They Work Together

File-based import supports the import of data from an external text or xml file to interface tables and then from interface tables to target application tables.

Overview of File-Based Data Import

File-based import includes the following:

- Source files with import data
- Import objects with available import attributes
• Mappings between source files and interface table columns

• Import Activities to define import options, a processing schedule, and monitor progress

**Source Files**

External data can be obtained in various ways and formatted in a text or xml file. The source file data is mapped to interface table columns using a Mapping. The source file is identified on an Import Activity, along with other import processing details. The file processing component of the file-based data import consists of reading the source file, parsing the data, and inserting the data into the appropriate interface tables.

**Objects**

Import objects are defined where interface tables exist and external files can be used to import data into the interface tables. Import Object definitions for Oracle objects that support file-based import are predefined and can be accessed with the appropriate security privilege. Individual object attributes represent the interface table columns and are used to map source file data or constant values in Mappings and Import Activity definitions. Use the Import Object definition to manage the display of attributes that can be mapped, to indicate required mappings, and to set site level default values as required.

**Mappings**

Import mapping enables you to predefine a mapping between the columns provided in a source file and the attributes pertaining to the objects being imported. Once you create a mapping, it can be reused in the Import Activity definition.

**Manage Import Activities**

An Import Activity definition provides the instructions for the import processing. It includes the source file or file location and mapping, plus import processing options and schedule. You can monitor the progress of the Import Activity processing and view completion reports for both successful records and errors.

**File-Based Import Processing: How it Works**

The file-based data import process includes processing the source file data and inserting it into the interface tables, moving the interface table data into the destination application tables, and then processing the attachments for the imported objects. Processing factors are subject to the settings defined for the Import Activity, Mapping, and Import Object. You can monitor the processing steps and view process reports for each Import Activity.

This topic describes the following:

• Inserting Data in the Interface Tables

• Interface Table Data Validation and Error Counts
• Interface Table to Destination Application Table Processing
• Importing Attachments
• Viewing Import Results

Inserting Data in the Interface Tables

Data exists in various sources and in various formats. The file import processing starts with reading the source data, parsing the data, and inserting into the appropriate interface tables. The source of the data comes from the following:

• Source file values mapped to target object attributes in the Import Activity.
• Constant values defined for target object attributes in the Import Activity.
• Default values defined for target object attributes in the Import Object.

Interface Table Data Validation and Error Counts

The data is initially validated against the predefined Import Mapping and the Import Object settings as the interface tables are being populated by the initial file import process. The interface table data is validated again before importing into the destination application tables.

• Validation includes:
  • Missing required values
  • Values that exceed the attribute length
  • Invalid values
  • Duplicates to existing records in the destination application tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping.

Note

For the Lead import object, the duplicate checking is only done for existing leads created within the look back days setting of the Import Activity.

• Duplicates to existing records in the destination application tables for Customer Data Management objects based on Matching Configurations.

• Errors

Most validation issues are recorded as errors, with the exception of Customer Data Management duplicates found during the Matching Configuration process. In this case, matched records are only considered as errors if:

• Customer Management Duplicates option is set to Do Not Import for the Import Activity and
The main object of the Import Activity is a consumer, customer, or legal entity object.

Allowable Error Count Threshold

The validation of the interface table occurs before any records are imported into the destination application tables. Once the validation process has completed, the count of records with errors is compared to the Allowable Error Count Threshold value specified for the Import Activity. A count above the threshold will stop the import process for all records. If the count is below the threshold, records without errors will import. In either case, records with errors will be reported in the Error and Exception files.

Interface Table to Destination Application Table Processing

The import process orchestrates the import for each of the component objects that make up the overall main objects of the Import Activity.

Importing Attachments

Once the objects have imported successfully, the attachments are processed. The import process matches the source file attachment name to the file name included in the compressed file entered on the Import Activity. The attachment file is imported into Universal Content Manager and then associated as an attachment to the imported object.

Viewing Import Results

You can monitor all file-based Import Activities that are currently scheduled to run, have completed successfully, or failed with errors. For each Import Activity, you can view the details pertaining to each underlying process. Once an Import Activity process has completed, the following processing reports are added as attachments to the process:

- Log file. Includes the records that were successfully imported plus the unique destination application table identifiers for the objects.
- Exception file. Includes the records that were not imported plus a reference to an error for each record that failed validation.
- Error file. Includes all the errors for each record that failed validation.

File-Based Import Objects: Explained

Import objects represent the application and attribute information for business objects that can be imported using external source files.

This topic describes the following:

- Import object management options
- Custom objects
Import Object Management Options

A single import object can have multiple associated components that are considered objects by themselves. An object and associated objects that can be imported within the same source file are grouped together within the application module class.

Note

Each object includes the Import Activity object (MktImpJobs1). The Import Activity object is a required component of the application module but is not mapped to a source file. All values for this object are derived from the Import Activity definition. Consequently, do not update the Map, Required, and Default Value settings for the Import Activity object.

The following table includes information about the import object:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes</td>
<td>A view-only listing of object attributes that represent each column in the interface table for the object.</td>
</tr>
<tr>
<td>Length</td>
<td>A view-only listing of widths for the columns in the interface tables. If the source file values for the attribute have more characters than the attribute length, the source file row will not be imported.</td>
</tr>
<tr>
<td>Default Value</td>
<td>Optionally, specify an attribute value to use if a value is not available from the source file or Import Activity constant value.</td>
</tr>
<tr>
<td>Map</td>
<td>Enable the list of attributes that can be mapped to a source file or constant value in the Import Mapping and Import Activity Map Fields step.</td>
</tr>
<tr>
<td>Required</td>
<td>Specify the list of attributes that must be mapped to source file columns. Consequently, if you have selected an attribute as required, you must also enable the Map option for that attribute. When mapping the external source file, the required target attribute defined for the object are displayed with an asterisk.</td>
</tr>
</tbody>
</table>

Custom Objects

To use the file-based import feature for custom objects, you must first generate the artifacts required for import. You generate these required artifacts within Oracle Fusion CRM Application Composer, after making your object model extensions.

File-Based Import Mapping: Explained

Import mapping enables you to predefined a mapping between the columns provided in a source file and the attributes pertaining to the objects being
imported. Once you create a mapping, it can be reused in the Import Activity definition.

This topic contains the following sections:

- Import options
- Source file options
- Target options

**Import Options**

The following attributes pertain to the import mapping.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The business object to be imported.</td>
</tr>
<tr>
<td>Name</td>
<td>The name that identifies the mapping in the Import Mapping and Import Activity UIs. If the mapping was initially created while mapping fields directly in the Import Activity user interface and automatically saved without providing a user-defined mapping name, the mapping name is derived from the Import Activity name and date.</td>
</tr>
<tr>
<td>Decimal Separator</td>
<td>The format of the fractional portion of numerical values in columns mapped to attributes with a decimal attribute type.</td>
</tr>
<tr>
<td>Date Format</td>
<td>The format of values in columns mapped to attributes with a date attribute type.</td>
</tr>
<tr>
<td>Timestamp Format</td>
<td>The format of values in columns mapped to attributes with a time stamp attribute type.</td>
</tr>
<tr>
<td>Lock</td>
<td>If selected, prevents any user, other than the creator of the mapping, from editing the mapping.</td>
</tr>
</tbody>
</table>

**Source File Options**

Map each column that the source file is expected to contain with a specific attribute.

The following table describes the details pertaining to columns provided in the source file:

<table>
<thead>
<tr>
<th>Source Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence</td>
<td>The sequence number in which the columns are expected to be provided in the source file. Two rows cannot have the same sequence number.</td>
</tr>
<tr>
<td>Column Name</td>
<td>The column name expected in the source file if a header row is included, or more generic values such as Column A, Column B, and so on, if the header row is not included for Text file types. The tagging structure is represented for XML file types.</td>
</tr>
<tr>
<td>Column Width</td>
<td>Use when the delimiter value is fixed width for Text file types only.</td>
</tr>
</tbody>
</table>
**Target Options**

The following table describes the details pertaining to corresponding attributes in the target application table:

<table>
<thead>
<tr>
<th>Target Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The group of import objects that represent the components of the business object being imported.</td>
</tr>
<tr>
<td>Attribute</td>
<td>The attribute name that represents the corresponding interface table column for the object.</td>
</tr>
<tr>
<td>Duplicate Validation</td>
<td>If selected, the attribute, along with other selected attributes, determines what constitutes a duplicate object when comparing objects in the interface tables and existing objects in the target application tables. For example, to validate the uniqueness of an object in the target application tables by the combination of an object's name and date, select <strong>Duplicate Validation</strong> for both attributes in the mapping.</td>
</tr>
</tbody>
</table>

**Import Activity Source File Options: Explained**

The Import Activity consists of a step by step guided process to assist you with creating an import activity for a given object.

This topic describes the source file options defined in the Import Activity that are used by the import process to locate and parse the source file data.

**Source File Data**

Enter attribute details pertaining to the source file as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Type</td>
<td>Source file must be either Text or XML.</td>
</tr>
<tr>
<td>Data Type, Delimiter, and Header Row Included</td>
<td>A Text file type can further be defined based on how the data is delimited and if the source file is expected to include a row of headings for each column.</td>
</tr>
<tr>
<td>Import Mapping</td>
<td>Displays a list of predefined mappings for the object selected for this import activity. The selected mapping will be used as the basis for mapping your source file in the next Import Activity step.</td>
</tr>
</tbody>
</table>

**Source File Location**

The following outlines the options that are available to you when locating your source file for import.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Selection</td>
<td>Select from the following file selections:</td>
</tr>
<tr>
<td></td>
<td>• Specific file</td>
</tr>
<tr>
<td></td>
<td>Enables you to upload a specific source file from a local file system, such as your desktop, a URL address, or from a network path. A file name is required for this option.</td>
</tr>
<tr>
<td></td>
<td>• Most recent file</td>
</tr>
<tr>
<td></td>
<td>Enables you to schedule repeating import activities without having to select a new file every time. This selection is only available when you select Network from the Upload From options.</td>
</tr>
<tr>
<td></td>
<td>You need to copy the new file to the specified network path for repeating import activities. You do not need to enter a file name for this option and can only upload your source file from a network path. The asterisk wildcard is supported for multiple characters. The question mark wildcard is supported for a single character.</td>
</tr>
<tr>
<td>Upload From</td>
<td>You can upload the source file from three locations:</td>
</tr>
<tr>
<td></td>
<td>• Desktop</td>
</tr>
<tr>
<td></td>
<td>• URL</td>
</tr>
<tr>
<td></td>
<td>• Network</td>
</tr>
<tr>
<td></td>
<td>If you select Desktop, a File Name field with an associated Update button is displayed. Click Update and browse to search for and select the file you want to upload.</td>
</tr>
<tr>
<td></td>
<td>If you select URL, enter the address location as in the following example format: <a href="http://www.example.com/">http://www.example.com/</a></td>
</tr>
<tr>
<td></td>
<td>If you select Network, enter the file name path as in the following example format: \ComputerName\SharedFolder\Resource\</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>If you selected the Specific File as your file selection option, then you will have to include the file name for both URL and Network file path locations.</td>
</tr>
</tbody>
</table>

**Import Activity Attachment Options: Explained**

Many objects can have multiple documents attached to it. The File Import Activity process allows you to import the documents attached to a specific object.
In the Summary section enter an import activity name and description and the primary object for which you want to import data from a source file. The list of objects displayed is controlled by your data security privileges. This topic describes how to define the attachment options pertaining to the import activity.

**Attachments**

As part of the file attachment import process setup, you must:

- Provide the relationship between the attachment file or files and the object in the source file with multiple columns each referencing a file name pertaining to the attachment.
- Provide file names in the source file columns for attachment corresponding to each object record (row)
- Select all the files associated with all the objects targeted in the current file import activity process
- Map the columns related to file names to specific object and attribute pertaining to the common attachment interface such as category, file name, file title and file description
- Monitor the process for uploading attachments that is activated as part of the file import activity process

**Selecting Documents**

You define the parameters for the import activity to include the primary object for which the data is included in the source file. If the object being imported has attachments, you will need perform an additional step of selecting documents that serve as attachments for each record being imported in the Attachments section. Select the Multiple Files option and then click on Browse to display the Universal Content Manager (UCM). From here you can select individual documents that serve as attachments or a single file that contains all these documents as follows:

- Select a pre-configured compressed file in Zip or Jar format that contains all the individual attachment documents. If the compressed file contains hierarchy of folders then the attachment import process will traverse through the hierarchy to search for specific file name.
- Select individual attachment documents which UCM automatically compresses into a Zip format. In this case, the individual document cannot be a compressed file.

Browse through the file system and select multiples files from across various folders. You must select all attachments in one operation. For example, you cannot select a few files now and then return later to select more attachments files.

**Import Activity Import Options: Explained**

The File Import Activity consists of a step by step guided process to assist you with creating an import activity for a given object.

This topic describes the import options defined in the Import Activity that are used by the import process to interpret source file data and import interface table data into the target application tables.

**Source File Data Transformation**

The following options are used to identify the formatting of source file data so the data can be correctly interpreted and transformed by the import process:
### Interface to Target Import Options

The following options are used when importing the interface table information to the target application tables:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Mode</td>
<td>Determines if the Import Activity process should create new records or update existing records.</td>
</tr>
<tr>
<td></td>
<td>If updating existing records, the record IDs must be provided in the source file. If an existing record is not found, a new record is created. Update mode is not supported for all import objects. Consequently, the Import Mode is set to Create and is not updatable for those objects.</td>
</tr>
<tr>
<td></td>
<td>If creating new records, the import process evaluates the data in the interface tables with existing objects in the target application tables for possible duplicates. Customer Data Management objects are evaluated using the rules defined in the set of Matching Configurations. All other objects are evaluated using the combination of attributes selected for duplicate validation in the predefined Import Mapping.</td>
</tr>
<tr>
<td>Allowable Error Count</td>
<td>An error count above the threshold will stop the import process for all records. If the error count is below the threshold, records without errors are imported. In either case, records with errors will be reported in the Error and Exception files. Validation errors include:</td>
</tr>
<tr>
<td></td>
<td>• Missing required values</td>
</tr>
<tr>
<td></td>
<td>• Values that exceed the attribute length</td>
</tr>
<tr>
<td></td>
<td>• Invalid identifiers and lookup codes</td>
</tr>
<tr>
<td></td>
<td>• Duplicates to existing records in the destination tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping</td>
</tr>
<tr>
<td></td>
<td>Duplicates found using matching configurations for Customer Data Management objects do not contribute to the error count.</td>
</tr>
<tr>
<td><strong>Notification E-Mail</strong></td>
<td>The e-mail of the intended recipient of import processing notifications.</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| **Customer Data Management Duplicates** | Consumer, customer, and legal entity objects imported by themselves or as components of another object are subject to duplicate verification. The duplicates are determined using the following matching configurations:  
  - Batch Location Basic Duplicate Identification  
  - Batch Person Basic Duplicate Identification  
  - Batch Organization Basic Duplicate Identification  
  
You can select from one of the following:  
  - **Do Not Import Duplicate Records**  
    If the main object of the Import Activity is a consumer, customer, or a legal entity object, rows that are matched to existing records will not be imported. These duplicates records are reported in the Exception and Error reports.  
    
    If the Customer Data Management objects are components of another object and one or more matches are found, the existing duplicate records are evaluated to determine the most recent record. The most recent record will be associated with the main object being imported.  
    
    For example, when importing a marketing response object, the consumer object is also a component of the response. If the consumer is matched to an existing record, the consumer in the interface tables is not imported. However, the response object will import and the most recent existing consumer record will be associated to the response.  
    
  - **Import Duplicate Records**  
    The Customer Data Management objects will be imported even if matched records exist.  
    
  - **Import Duplicate Records and Create Resolution Request**  
    The Customer Data Management objects will be imported even if matched records exist. In addition, a duplicate resolution request is created and displayed in the Customer Data Management, Duplicate Resolution work area. |
Duplicate Look Back Days

This option applies only to the Lead import object. Only existing leads created within the period determined by the look back days value are evaluated for duplicates based on the attributes selected for duplicate validation in the predefined import mapping. If a duplicate is found, the lead will not be imported and the duplicate record will be reported on the Exception report. Duplicate leads are included in the calculation of the allowable error count threshold.

Import Activity Field Mapping: Explained

After entering your import options, the second step of the import activity process is to map fields in the source file to the corresponding target attributes.

This topic explains:

- Map Fields
- Saving the Import Mapping
- Constant Values

Map Fields

The Map Fields section can be subdivided into source file columns and target attribute columns.

The source column header value is derived from one of the following:

- Predefined mapping, if one is selected
- The source file, if the Header Row Included option is selected in the first step of the Import Activity definition (for Text file type only)
- Generic values of Column A, Column B, and so on, if the Header Row Included option is not selected (for Text file type only)
- XML tagging structure (for XML file type only)

The following table outlines the source columns:

<table>
<thead>
<tr>
<th>Source Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Header</td>
<td>Represents the column header for Text file types and the tagging structure for XML file types.</td>
</tr>
<tr>
<td>Example Value</td>
<td>Values are derived from the first source file saved with the predefined mapping. If you did not select a predefined mapping, the example values are taken from the first data row in the source file selected in the first step of the Import Activity definition.</td>
</tr>
<tr>
<td>Ignore</td>
<td>Select this option if you do not want to import the source file data in that column.</td>
</tr>
</tbody>
</table>

The following table outlines the target columns:

<table>
<thead>
<tr>
<th>Target Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>The group of import objects that represent the components of the business object being imported.</td>
</tr>
</tbody>
</table>
**Attribute**
The attribute name that represents the corresponding interface table column for the object.

**Saving the Import Mapping**
The mapping between source file information and target attributes is saved as a reusable mapping when the Import Activity is saved, using the import activity name and date to derive a mapping name. If you selected a predefined mapping, modifications made in the Import Activity to an unlocked mapping will update and save to the predefined mapping. If the predefined mapping is locked, a modified mapping will be saved as a new mapping. To specify a mapping name for new mappings, select the **Save As** option from the Map Fields **Actions** menu.

**Constant Values**
Constant values provide a way to specify a value for a target attribute that all imported objects will inherit. For example, if a source file does not contain a column for business unit and all of the objects in the file belong to the same business unit, enter a constant value for the object and business unit attribute.

**File-Based Import Monitoring: Explained**
You can monitor all file import activities that are currently scheduled to run, have completed successfully, or failed with errors. For each import activity, you can view the details pertaining to each underlying process and make necessary updates for any failed records to import again.

You can view the list of import activities from the Manage Import Activities page. Select the import activity that you want to monitor by clicking on the hyperlink in the corresponding Status column. The View Import Status results page is displayed which contains the following sections:

- **Files Processed**
- **Import Processes**

**Files Processed**
The Files Processed section displays a row for each source file that is processed.

The import processing details are summarized and displayed for each source file and include the following:

<table>
<thead>
<tr>
<th>File Processing Summary Information</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records Read From File</td>
<td>The number of records read from the source file.</td>
</tr>
<tr>
<td>Format Errors</td>
<td>The number of errors found when processing data to insert into the interface tables from the source file, Import Activity constants, and Import Object value default values. View the error details in the Exception and Error files attached to the process.</td>
</tr>
<tr>
<td>Load Errors</td>
<td>The number of errors found when importing data from the interface tables to the destination application tables. View the error details in the Exception and Error files attached to the process.</td>
</tr>
</tbody>
</table>
Successfully Loaded | The number of import objects imported to the application destination tables. If the import object is made up of multiple components, each component is counted as successfully loaded. Consequently the Successfully Loaded count may be larger than the Records Read From File count. View the successful record details in the Log file attached to the process.

Attachments | Once an Import Activity process has completed, processing reports are included in the Attachments column. The Log file includes the records that were successfully imported plus the unique destination application table identifiers for the objects. The Exception file includes the records that were not imported plus a reference to one of the errors for each record that failed. The Error file includes all the errors for each record that failed validation.

**Import Processes**

From the Import Processes section, you can view details pertaining to each process involved in importing the objects in the source file. A listing of brief messages provides information on processing steps within each underlying process.

**Importing Notes: Explained**

You can create new notes or update existing notes by importing data through interface tables. To populate the interface table, you provide the data details in a source file and use the file-based import feature. Having a good understanding of the import entity, interface table, and destination table will help you prepare your import data.

Consider the following when importing notes:

- File-based import
- Import object entity, interface table, and destination table

**File-Based Import**

The file-based import process reads the data included in your XML or text file, populates the interface table, and imports the data into the application destination table. The File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the note import object, create source file mappings, and schedule the import activities.

You can import attachments to your import object by including the file names in your note source file and selecting the files when defining your import activity.

**Import Object Entity, Interface Table, and Destination Table**

The note import object consists of one entity that forms the note. Since interface tables are tied to import object entities, there is only one interface table for importing notes. You can map your source file data to the import entity attributes that correspond to the interface table columns. The import activity process will populate the interface table based on the mapping and your source file. If you need the unique IDs of existing application data for your import
data, use the Define Data Export Setup and Maintenance task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the FusionApps: Interface Table asset type.

The following lists the object entity, tables, and resulting application object:

<table>
<thead>
<tr>
<th>File-Based Import Entity</th>
<th>Interface Table</th>
<th>Destination Table</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoteImport</td>
<td>ZMM_IMP_NOTES</td>
<td>ZMM_NOTES</td>
<td>Note</td>
</tr>
</tbody>
</table>

**Importing Tasks: Explained**

You can create new tasks by importing data through interface tables. To populate the interface table you provide the data details in a source file and use the file-based import feature. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing tasks:

- File-based import
- Import object entities, interface tables, and destination tables

**File-Based Import**

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the task import object, create source file mappings, and schedule the import activities.

**Note**

If using a text file, one row of data in your source file represents a single task. Consequently, if you have a one-to-many scenario for some of the task’s details, you must repeat each set of data in the source file on the same row. For example, if the tasks in your source file typically include up to two contacts, create a set of data for contact one and an additional set of data for contact two to include on the same row as your task.

You can import attachments to your import object by including the file names in your task source file and selecting the files when defining your import activity.

**Import Object Entities, Interface Tables, and Destination Tables**

The task import object consists of entities that form the task. Since interface tables are tied to import object entities and the task object consists of many entities, there are multiple interface tables included in importing tasks. You map your source file data to import entity attributes that correspond to the
interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If you need the unique IDs of existing application data for your import data, use the **Define Data Export Setup and Maintenance** task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the **FusionApps: Interface Table** asset type.

The following table lists the object entities, tables, and resulting application objects:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaskImport</td>
<td>ZMM_IMP_WF_TASK</td>
<td>CRM_FUSION_SOAINFR</td>
<td>Task</td>
</tr>
<tr>
<td>TaskAssigneeImport</td>
<td>ZMM_IMP_WF_ASSIGNE</td>
<td>WF_ASSIGNEES</td>
<td>Task Assignee</td>
</tr>
<tr>
<td>TaskContactImport</td>
<td>ZMM_IMP_TASK_CONT</td>
<td>ZMM_ACT_CONTACTS</td>
<td>Task Contact</td>
</tr>
<tr>
<td>NoteImport</td>
<td>ZMM_IMP_NOTES</td>
<td>ZMM_NOTES</td>
<td>Task Note</td>
</tr>
</tbody>
</table>

**Importing Interactions: Explained**

You can create new interactions by importing data through interface tables. To populate the interface tables, you provide the data details in a source file and use the file-based import feature. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing interactions:

- File-based import
- Import object entities, interface tables, and destination tables

**File-Based Import**

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The **File-Based Data Import Setup and Maintenance** task list includes the tasks needed to configure the interaction import object, create source file mappings, and schedule the import activities.

**Note**

If using a text file, one row of data in your source file represents a single interaction. Consequently, if you have a one-to-many scenario for some of the interaction’s details, you must repeat each set of data in the source file on the same row. For example, if the interactions in your source file typically include up to two contacts, create a set of data for contact one and an additional set of data for contact two to include on the same row as your interaction.
Import Object Entities, Interface Tables, and Destination Tables

The interaction import object consists of entities that form the interaction. Since interface tables are tied to import object entities and the interaction object consists of many entities, there are multiple interface tables included in importing interactions. You can map your source file data to the import entity attributes that correspond to the interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If you need the unique IDs of existing application data for your import data, use the Define Data Export Setup and Maintenance task list to export the information.

Note

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the FusionApps: Interface Table asset type.

The following table lists the object entities, tables, and resulting application objects:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>InteractionImport</td>
<td>ZMM_IMP_INT_INTERACTIONS</td>
<td>ZMM_INTER_INTERACTIONS</td>
<td>Interaction</td>
</tr>
<tr>
<td>InteractionAssociationImp</td>
<td>ZMM_IMP_INT_ASSOCIATIONS</td>
<td>ZMM_INTER_ASSOCIATIONS</td>
<td>Interaction associations</td>
</tr>
<tr>
<td>InteractionParticipantImp</td>
<td>ZMM_IMP_INT_PARTICIPANTS</td>
<td>ZMM_INTER_PARTICIPANTS</td>
<td>Interaction resources and contacts</td>
</tr>
</tbody>
</table>

Importing Appointments: Explained

You can create new appointments and update existing appointments by importing data through interface tables. To populate the interface table, you provide the data details in a source file and use the file-based import feature. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing appointments:

- File-based import
- Import object entities, interface tables, and destination tables

File-Based Import

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the appointment import object, create source file mappings, and schedule the import activities.

Note
If using a text file, one row of data in your source file represents a single appointment. Consequently, if you have a one-to-many scenario for some of the appointment’s details, you must repeat each set of data in the source file on the same row. For example, if the appointments in your source file typically include up to two contacts, create a set of data for contact one and an additional set of data for contact two to include on the same row as your appointment.

You can import attachments to your import object by including the file names in your appointment source file and selecting the files when defining your import activity.

**Import Object Entities, Interface Tables, and Destination Tables**

The appointment import object consists of entities that form the appointment. Since interface tables are tied to import object entities and the appointment object consists of many entities, there are multiple interface tables included in importing appointments. You can map your source file data to the import entity attributes that correspond to the interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If you need the unique IDs of existing application data for your import data, use the **Define Data Export Setup and Maintenance** task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the **FusionApps: Interface Table** asset type.

The following lists the appointment object entities, tables, and resulting application objects:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppointmentImport</td>
<td>ZMM_IMP_ACTIVITIES</td>
<td>ZMM_ACTIVITIES</td>
<td>Appointment</td>
</tr>
<tr>
<td>ActivityAssociationImport</td>
<td>ZMM_IMP_ACT_ASSOCI</td>
<td>ZMM_ACT_ASSOCIAT</td>
<td>Business object related to the appointment</td>
</tr>
<tr>
<td>ActivityAssigneeImport</td>
<td>ZMM_IMP_ACT_ASSIGN</td>
<td>ZMM_ACT_ASSIGNMEN</td>
<td>Appointment participants</td>
</tr>
<tr>
<td>ActivityContactImport</td>
<td>ZMM_IMP_ACT_CONTACT</td>
<td>ZMM_ACT_CONTACTS</td>
<td>Appointment contact</td>
</tr>
<tr>
<td>NoteImport</td>
<td>ZMM_IMP_NOTES</td>
<td>ZMM_NOTE</td>
<td>Appointment note</td>
</tr>
</tbody>
</table>

**Importing Promotions: Explained**

You can create new promotions by importing data through interface tables. There are two options for populating the interface tables: using the tool of your preference to load the data or an automated pull from a data file. If you plan to provide the data details in a source file, use the file-based import feature. If you will populate the interface tables directly, use scheduled processes to import the data. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.
Consider the following when importing promotions:

- File-based import option
- Scheduled process import option
- Import object entities, interface tables, and destination tables

File-Based Import Option

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the promotion import object, create source file mappings, and schedule the import activities.

Note

If using a text file, one row of data in your source file represents a single promotion. Consequently, if you have a one-to-many scenario for some of the promotion’s details, you must repeat each set of data in the source file on the same row. For example, if the promotions in your source file typically include up to two coupons, create a set of data for coupon one and an additional set of data for coupon two to include on the same row as your promotion.

Scheduled Process Import Option

Navigate to Scheduled Processes, after you have populated the interface tables, to schedule the import of data from the interface tables to the destination tables.

The following displays the process you can schedule to import promotions and related information:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Process Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BulkImportProcess_PROMOTION_LOAD_ALL</td>
<td>Import Promotions and Associated Coupons</td>
</tr>
</tbody>
</table>

Import Object Entities, Interface Tables, and Destination Tables

The promotion import object consists of entities that form the promotion. Since interface tables are tied to import object entities and the promotion object consists of many entities, there are multiple interface tables included in importing promotions. If you are using file-based import, you can map your source file data to import entity attributes that correspond to the interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If using scheduled processes, populate the tables directly using your preferred tool. If you need the unique IDs of existing application data for your import data, use the Define Data Export Setup and Maintenance task list to export the information.

Note

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the FusionApps: Interface Table asset type.
The following table lists the object entities, tables, and resulting application objects:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>PromotionBulkImport</td>
<td>MOP_IMP_PROMOTIONS</td>
<td>MOP_PROMOTIONS_B</td>
<td>Promotion</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MOP_PROMOTIONS_TL</td>
<td></td>
</tr>
<tr>
<td>PromotionCouponBulkIm</td>
<td>MOP_IMP_PROMO_COUPONS</td>
<td>MOP_PROMO_COUPON</td>
<td>Promotion Coupon</td>
</tr>
</tbody>
</table>

**Importing Product Groups: Explained**

You can create new product groups by importing data through interface tables. There are two options for populating the interface tables: using the tool of your preference to load the data or an automated pull from a data file. If you plan to provide the data details in a source file, use the file-based import feature. If you will populate the interface tables directly, use scheduled processes to import the data. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing product groups:

- File-based import option
- Scheduled process import option
- Import object entities, interface tables, and destination tables

**File-Based Import Option**

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The **File-Based Data Import Setup and Maintenance** task list includes the tasks needed to configure the product group import object, create source file mappings, and schedule the import activities.

**Note**

If using a text file, one row of data in your source file represents a single product group. Consequently, if you have a one-to-many scenario for some of the product group’s details, you must repeat each set of data in the source file on the same row. For example, if the product groups in your source file typically include two associated products, create a set of data for product one and an additional set of data for product two to include on the same row as your product group.

You can use the same source file to import both extensible custom attributes and the standard product group object attributes. First, design your object model extensions in Oracle Fusion CRM Application Composer and generate the required artifacts to register your extensions and make them available for importing. When complete, the import object is updated with the extensible attributes, which can then be mapped to your source file data.

**Scheduled Process Import Option**

Navigate to **Scheduled Processes**, after you have populated the interface tables, to schedule the import of data from the interface tables to the destination tables.
You can import extensible, custom attributes in the same process as your product group object. Design your object model extensions in Oracle Fusion CRM Application Composer and generate the required artifacts to register your extensions and make them available for importing before you populate the corresponding extensible columns in the interface tables.

The following displays the process you can schedule to import product groups and related information:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Process Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BulkImportProcess_PRODUCT_GROUPS_MATCH</td>
<td>Evaluate Product Groups and Related Content for Import</td>
</tr>
</tbody>
</table>

**Import Object Entities, Interface Tables, and Destination Tables**

The product group import object consists of entities that form the product group. Since interface tables are tied to import object entities and the product group object consists of many entities, there are multiple interface tables included in importing product groups. If you are using file-based import, you can map your source file data to import entity attributes that correspond to the interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If using scheduled processes, populate the tables directly using your preferred tool. If you need the unique IDs of existing application data for your import data, use the **Define Data Export Setup and Maintenance** task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the FusionApps: Interface Table asset type.

The following table lists the object entities, tables, and resulting application objects:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductGroupBulkImport</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>Product group</td>
</tr>
<tr>
<td>ProductGroupItemBulkIm</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>Product group products</td>
</tr>
<tr>
<td>ProductGroupRelationBulkIm</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>QSC_IMP_PROD_GROUP</td>
<td>Related product groups</td>
</tr>
</tbody>
</table>

**Importing Marketing Budgets: Explained**

You can create new or update existing marketing budgets by importing data through interface tables. To populate the interface table you provide the data details in a source file and use the file-based import feature. Having a good understanding of the import entities, interface tables, and destination tables will help you prepare your import data.
Consider the following when importing marketing budgets:

- File-based import
- Import object entities, interface tables, and destination tables

**File-Based Import**

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The **File-Based Data Import Setup and Maintenance** task list includes the tasks needed to configure the marketing budget import object, create source file mappings, and schedule the import activities.

You can use the same source file to import both extensible custom attributes and the standard marketing budget object attributes. First, design your object model extensions in Oracle Fusion CRM Application Composer and generate the required artifacts to register your extensions and make them available for importing. When complete, the import object is updated with the extensible attributes, which can then be mapped to your source file data.

Marketing budget notes are imported independently using a separate source file and import activity. Include the data that will identify the associated budget in the note’s source file.

**Note**

To obtain unique IDs for the marketing budgets you have just imported, you can view the log file of successful records in the budget’s import activity by navigating to **Setup and Maintenance** and selecting the **Manage File Import Activities** task. When the budget import activity is complete, select the **Schedule Status** link to access the log file containing the budget IDs.

You can import attachments to your import object by including the file names in your marketing budget source file and selecting the files when defining your import activity.

**Import Object Entities, Interface Tables, and Destination Tables**

The marketing budget import object consists of one entity that forms the budget. Since interface tables are tied to import object entities, there is only one interface table for importing budgets. You map your source file data to import entity attributes that correspond to the interface table columns. The import activity process will populate the interface tables based on the mapping and your source file. If you need the unique IDs of existing application data for your import data, use the **Define Data Export Setup and Maintenance** task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching on a specific interface table name or initiating a search using the **FusionApps: Interface Table** asset type.

The following table lists the object entity, tables, and resulting application object:
FAQs for File-Based Data Import

What determines the list of objects displayed?

A single import object can have multiple associated components that are considered objects by themselves. Whether or not an associated object can be grouped as a component of another object for the purpose of file import is determined by the complexity of the object structure and how it is stored in the data model. Oracle Fusion provides import objects predefined to meet the file processing import requirements. Consequently, in some cases, more than one source file may be required to capture all associated components of an object.

What happens if I inactivate an Import Activity?

The Import Activity will not stop the currently running process. However, it will stop the next process that has not started plus any future repeating file import activities. You can always activate the process at a later stage.

What happens if I add a marketing list in the Import Activity definition?

File-based data import enables you to record consumers and organization contacts in a marketing list when importing consumer, lead, and response import objects. Select an existing list or create a new one. A marketing list is assigned the list type value of Imported if created while defining an import activity. After the objects are imported successfully, the consumers and contacts are added as members of the marketing list.

Define Data Export

Bulk Export: Overview

The Bulk Export application provides a mechanism to extract large volumes of data from Fusion CRM objects. These extracts can be the full set of records for an object or incremental extracts. For example, data extracted for a specific period of time, from the hosted CRM system to an on-premise database that resides behind a user's fire-wall. The system will create comma separated variable or tab delimited files with the extracted data, which will be available to users as attachments to the batch records that have been executed.

The following figure depicts the process of selecting data for export, scheduling and finally delivering the exported data file.
Bulk Export Process Definition: Explained

This solution provides a mechanism to extract large volumes of data from Fusion CRM objects, both as extracts of a full set of records for an object as well as incremental extracts. The system will create comma or tab delimited files with the extracted data which will be available to users as attachments to the batch records that have been executed.

In order to create the extracts, two steps must be completed. First, mapping files for the full and incremental extract processes must be defined in the Fusion CRM system. These maps will specify which columns and filters will be applied to each export process for each export object. For the incremental extracts, filters can be created that leverage time stamps to determine which rows will be queried out of the system. All mapping files will be saved in the system and reused for each extract.
Next, the hourly and weekly data export processes are scheduled in the Fusion export tool. For any required incremental and scheduled export, the export task should either exist or created through the UI. Oracle Web Services would only be used to schedule the export and start it. After each export process executes and completes, a comma or tab delimited data file will be created and stored in the Fusion system as an attachment. The formatted file can be downloaded by using the `getAttachment()` web service or by using the interactive UI in the export tool.

There are no transactional steps for this process in the Fusion CRM application, there are only prerequisite setup steps. Once these steps are complete the process should run automatically. The prerequisite steps in Fusion are to create an export map and export job schedule for each object to be extracted (this only needs to be done once).

The Bulk Export Process Definition is made up of the Export Map and the processing schedule. See the steps below.
Select the Export Object

The export object is the Fusion database object where the data resides. It is made up of attributes. If you need to export data from a custom table, you must register the object as an export object. This is accomplished from the Manage Export Process UI, Manage Export Objects action. All the delivered tables and their attributes are available for export.

Select Attributes

The export object is made up of attributes. These attributes may be selected for export or not included. You can edit the header text of the attribute to make its meaning more clear to other users of this process.

Filter Criteria

Each attribute may have limits or conditions enforced. Various operators are available for selecting the data to precisely select the data required for the export. You can save the filter criteria and then modify the criteria and save it under a new name. You can then change the filter by coming here to select an alternate filter name. Because the filters are related to the export object, if you reuse a map and change the filter, you are changing it for any Export Process Definition that uses that map. The attributes you use for the map have no bearing on what is available in the filter. All fields from the VO are available for use in the filter. For example, you can filter by TYPE but not show TYPE in the output.

Schedule the Export Process

Once defined, the export process is scheduled. You can run the process immediately or at the time and date of your choosing. If you decide to schedule the job at a later date you can also choose to set up a recurring schedule of extracts.

Activate the Job

By clicking on the Activate button, you make the job available to be run. It does not start an export process.

Bulk Export Process Components: How They Work Together

In the two step process used by Oracle Fusion Bulk Export, the first is the mapping of files for the full and incremental extract processes. The second step is the scheduling of the export. You create a process definition that includes both of these steps.

The process definition has three components that together make exporting data easier by leveraging the export maps that you have already built. The process name, the export process ID and the export map ID all serve to identify the
specific process definition as well as leverage your work with reusable export maps

**Process Name**
A user-supplied, natural language way to refer to the Export Process Definition. This enables you to refer to the export process definition easily rather than using the machine generated ID. For example, use Customer or some other meaningful name as the export process name instead of the export process ID 100000019897192.

**Export Process ID**
A unique, system generated identifier for the export process definition that ties together the export map, with its export objects and filters, and the defined export schedule.
Export Map ID

A unique identifier for the export map itself. You can name the export map or
leave the field blank for a system generated map name to be entered. You can
reuse the export map in different process definitions. For example, you could
create a process definition to export all the data from the Customer export object.
You could then reuse that export map and apply a new filter on the data to create
an incremental export, such as data accrued since the last export date.

FAQs for Define Data Export

How can I tell which objects to select?

Review the requirements for the data to be exported and determine the source
view object that holds the attributes you want.

How can I create a subset of data for export?

Full sets of data are not always required for export. To create a subset of data, use
filter criteria to determine the time frame or scope of data, based on values of the
attributes. For example, to find activities for a certain period, use a project start
date from 1/1/11 through 3/31/11, navigate to the Export Objects Detail Sub
Page and click the filter icon. Fill in the filter criteria dialog for the project start
dates to select the data to be exported. You run the export by navigating to the
Setup and Maintenance menu, selecting Manage Task Lists and Tasks. Then,
search for Schedule Export Processes and click the Go to Task icon on the line
for this task.

What happens if I need data from more than one view object?

Select as many view objects as required to be export objects for the export
process. Choose the individual attributes required from each export object.

How can I see my exported data?

You can look on the Schedule Export Processes, Overview page to see the
History subpage. The column Exported Data File shows a hyperlink to your
output file. This file will be a comma separated variable or a tab delimited file.
Click that link to open the file and see the exported data.

How can I use my own View Object for export?

You can use your own defined view objects as a source for Bulk Export. To
register your view objects for export, select Setup and Maintenance from the
Tools menu and search for the Manage Export Objects task. Click the Go to
Task icon and on the Manage Export Objects page click the Create icon to add
your View Object, making it available for use.

What happens if I change the sequence number or header text?

Changing the sequence number changes the order of the attributes in the
exported data file. Changing the header text enables you to give a more intuitive
meaning to the attribute and the associated data.
Manage Calendar Profile Option

Creating the CRM Common Calendar: Worked Example

The Oracle Fusion Customer Relationship Management (CRM) common calendar is used across CRM applications. The calendar utilizes the Accounting Calendar Default profile option that is not set when the delivered product is installed. First, create an accounting calendar with calendar periods appropriate for your CRM needs, and give it a unique name, CRM Calendar, for example. Then, you must specify that calendar in the Accounting Calendar Default profile option. Use the tasks noted below in the Set and Maintenance area to accomplish these tasks.

Creating the Accounting Calendar

1. In the Set Up and Maintenance area, Overview page, All Tasks tab, search for the topic names containing Accounting Calendar

2. Manage Accounting Calendars will show up in the results box. Click the Go To Task icon.

3. On the Managing Accounting Calendars page, click the Create icon.

4. Fill in the required fields as appropriate for the calendar and click Next. Make note of the name utilized as this will be required in the next topic.

   Note

   When creating the calendar, the first calendar date should be the first date of the period of the oldest historical data on which you will be reporting. For example, if you were to select January 1, 2010 as your first calendar date, you would only be able to enter or import historical data associated with this date and beyond.

5. Insure that the calendar data is correct and click Save.

Managing the CRM Common Calendar Profile Option

The CRM calendar profile option needs to be associated with the new accounting calendar. Follow these steps:

   Note

   While the Common Financial Calendar feature of Oracle Fusion Applications supports creation of more than one calendar, Fusion CRM may only be associated with one calendar. Many features of Fusion CRM utilize this common calendar profile option and changing it could result in the loss of data for one or more applications. Oracle strongly recommends that you do not change the selected Accounting Calendar Default (ZCA_COMMON_CALENDAR) profile option calendar value once it is set.

1. Navigate to the Set Up and Maintenance, Overview page, All Tasks tab, and search for task names containing Calendar.
2. Locate Manage Calendar Profile Option and click Go To Task.

3. On the Manage Calendar Profile Option page, locate the ZCA_COMMONCALENDAR: Profile Values sub page and click the one profile option value line.

4. Click the drop down for Profile Value and select the calendar name that you created above. Click Save and Close.

Define Sales Prediction Configuration

Sales Prediction Engine: Overview

Predictive models analyze sales data to evaluate buying patterns and selling win or loss rates. The model can be used to identify customer profiles with a greater likelihood to buy certain target products. After evaluation of model results, lead generation can be scheduled in order to disseminate lead recommendations to users who can benefit from the insight gathered. Each lead recommendation includes win likelihood in addition to the average expected revenue and sales cycle duration from past similar deals for the same product to similar customers.

Selecting Model Entities and Attributes: Examples

Oracle Fusion Sales Prediction Engine leverages the power of predictive analytical models to identify patterns and correlation of data for the purpose of identifying what products to consider positioning next to your customers. The application leverages multiple mathematical models in order to formulate the likelihood a given customer will purchase a specific product, the estimated revenue which can be expected, and the duration of the estimated sales effort.

After the statistical model generates against the historical sales data based on the selected entities and attributes, summary and detail reports show critical insights as to what customers buy and can be used to predict the right products for the right customers. You can then use these insights to refine the process of generating leads based on what the customers are more likely to buy. Moreover, the same model can be used to predict the win likelihood of current opportunity revenue based on analysis of similar opportunities in the past. Also, product domain or market experts can write prediction rules to recommend products based on a set of rules conditions, utilizing all available customer profile attributes as well as other metrics.

The statistical analysis will identify which data has an influence in determining a likelihood to buy. After the statistical model generates against the historical sales data based on the selected entities and attributes, summary and detail reports show critical insights as to what customers buy and can be used to predict the right products for the right customers. As such, the decision regarding the selection of entities and attributes is critical. While the selection of certain entities and attributes may seem logical based on the awareness of sales behavior (customers in certain industries have a stronger affinity for certain products, for example), where there is uncertainty, the statistical model analysis will provide the necessary insight into whether patterns and correlation emerge. As such, as much available data as possible should be leveraged for the purpose of
evaluation. Some factors which weight into the decision include the availability and accuracy of the data.

Additionally, the application allows for the inclusion of expert insight from product management and sales and marketing operations. These expert insights can be captured via prediction rules. The same data made available to and leveraged by the predictive model, is also available for rule authoring.

**Scenario**

Your company sells a service that appeals mostly to larger companies, and another service that targets smaller customers. If a customer purchased one of your product packages, then the customer already has all service needs covered by the package. You want to know, given a product recommendation, if credit score, asset, and customer size are important predictors when it comes to recommending this particular product.

You select the following entities and attributes:

- Customer Profile
- Annual Revenue
- Credit Score
- Customer Size Code
- Past Purchased Products or Services
- Assets and Service Contracts

These entities and associated attributes entail the data available to the enterprise which the predictive models can evaluate for identifying correlation, and which you can use to create prediction rules. Over time, you can further refine the selections based on availability of data and the cost to integrate that data for evaluation.

**Define Outlook Integration**

**Implement Oracle Fusion CRM for Microsoft Outlook: Explained**

Before using the Oracle Fusion CRM for Microsoft Outlook application, several setup tasks must be performed. Some of these are Fusion-specific tasks that are done by the environment hosting team or the customer implementation team. Other tasks are related to setting up the users’ computers to use the application, including the install and initialization of the extensions to Microsoft Outlook (Outlook). These tasks are described in more detail in the sections that follow.

For information on supported software versions, see the related topic, Supported Software for Oracle Fusion CRM for Microsoft Outlook: Explained.

**Overview of Fusion-specific Setup Tasks**

At a high level, the following are the Oracle Fusion-specific setup tasks involved in implementing CRM for Microsoft Outlook:
• Required: Install Fusion CRM, including the CRM for Microsoft Outlook application.

• Required: Perform Fusion setup tasks for Oracle Fusion Common Components, Oracle Fusion Customer Center, Oracle Fusion Sales, and Oracle Fusion Marketing.

• Optional: Perform customization and security changes for CRM for Microsoft Outlook, after initial setup.

**Overview of Setup Tasks Required for Each Computer Running CRM for Microsoft Outlook**

At a high level, the following are the setup tasks required for each computer that will run CRM for Microsoft Outlook:

• Required: If not already present, install Microsoft .NET framework version 3.5 SP1 (or later).

• Required: Download and install the Fusion CRM server certificate.

• Required: Download and run the CRM for Microsoft Outlook installer.

• Required: Complete First Run Assistant to set up application options and perform initial synchronization to get Outlook configuration and user data from the Fusion CRM application.

**Implementation Task Flow Overview**

The overall process flow for implementing CRM for Microsoft Outlook is shown in this section.
Fusion-specific Implementation Tasks

Following are the CRM for Microsoft Outlook implementation tasks specific to Oracle Fusion.

1. Install Oracle Fusion CRM Applications suite

   As a prerequisite setup task, provision the server environment and install the Fusion CRM Applications suite. This task is typically completed by the hosting operations team or customer implementing the Oracle Fusion CRM Applications suite and is the basis for the rest of the setup steps described in this section.

2. Perform CRM setup tasks for functionality used by CRM for Microsoft Outlook

   Because CRM for Microsoft Outlook allows users to access and manage their CRM data in Microsoft Outlook, it is necessary to complete the required setup tasks for the relevant CRM functionality. For example, the following setup tasks must be completed before using CRM for Microsoft Outlook:

   • Set up reference data, such as: address and phone formats, currencies, geographies, and resources.
   • Set up CRM functional areas exposed in CRM for Microsoft Outlook, such as: calendar and task management, customer and contact management, lead management, and opportunity and revenue management, including the sales product catalog.

3. Optionally, CRM for Microsoft Outlook can be configured by completing these Outlook-specific setup tasks:

   • Configure CRM for Microsoft Outlook client configuration files: Configure only if Outlook client customizations are needed
   • Configure CRM for Microsoft Outlook client deployment packages: Configure only if Outlook client customizations are needed
   • Configure CRM for Microsoft Outlook server configuration file: Configure only if Outlook configuration includes references to new services

Other, security-related tasks, performed in Oracle Fusion Authorization Policy Manager (APM), may be necessary depending upon your applications configuration. Perform these tasks after initial setup, as needed. If new job roles are created, you will need to associate these new roles with the predefined data privileges and Outlook configuration packages. If you create custom Outlook deployment packages, there are additional steps required. See the "Related Topics" section at the end of this topic for more information.

Non-Fusion Implementation Tasks

Following are the non-Fusion implementation tasks for CRM for Microsoft Outlook.

• Verify Microsoft .NET Framework 3.5 SP1 or higher is installed on all computers that run CRM for Microsoft Outlook.
• Verify each user has a Microsoft Exchange mail profile configured withCached Exchange Mode (which supports offline storage in an .OST file format) or has a separate personal folders storage (in .PST file format) to store CRM data.

• Deploy the Fusion public certificate into users' Personal and Trusted Root Certificate Authorities directories on users' computers. The certificate is provided by the environment hosting team or the group implementing Fusion CRM. See the related topic, Options for Deploying the Public Certificate: Explained, for steps describing how users can import the certificate themselves or how to automate the process.

• Verify that each user can access the CRM for Microsoft Outlook installer from the download page in the Sales application. The download page is accessible from the application preferences menu.

**Run CRM for Microsoft Outlook Installer**

Each user must run the CRM for Microsoft Outlook installer on his/her computer. See the related topic, Deploying and Installing Oracle Fusion CRM for Microsoft Outlook: Explained, for more information.

**Creating Deployment Packages: Explained**

In Oracle Fusion CRM for Microsoft Outlook, deployment packages contain metadata files that describe the CRM application extensions deployed to users’ computers. To provide users access to a new client configuration, you can either create a new deployment package or create a new instance of an existing package, as discussed in the following sections.

**Create New Deployment Package**

When you create a new package, in addition to activating it, you must configure a data security policy that allows users to access the package. This secondary task is done in Oracle Fusion Authorization Policy Manager (APM) and involves the following steps:

1. In the top left section of the APM application window, use global search to search for Database Resources using search criteria equal to Outlook. This should return the result, Outlook Edition Metadata Package.

2. Select the Edit button on the Search Results pane to edit the Outlook Edition Metadata Package database resource.

3. In the Edit Database Resource tab, select the Condition tab and create a new condition on the database resource. Specify any unique name/display name, and set the SQL predicate to `package_name = '<name_of_deployment_package>'` (for example, `package name = 'NewOutlookPackage'`).

4. Select the Submit button to commit the change.

5. Repeat step 2. In the search results pane, select Edit to reopen the Edit Database Resource page to edit the Outlook Edition Metadata Package database resource.

6. In the Edit Database Resource tab, select the Policy tab, and select the policy that should have access to the new package (for example, `ZOE_SALES_MGR_OUTLOOK_DUTY`), and then select Edit.
7. In the lower section of the page, select the Rule tab.

8. Select the lookup control next to the condition field and select the new condition created in step 3.

9. Select Submit to commit the changes.

Create New Instance of Existing Package

When you use an existing package, you create a new instance of the package with different configuration files. When using this method, you must inactivate the previous instance and activate the new instance. There is no need to configure a data policy when creating a new instance of an existing deployment package.

Oracle Fusion CRM for Microsoft Outlook and the Fusion Server: How They Fit Together

Oracle Fusion CRM for Microsoft Outlook is a composite application that allows users to work with Oracle Fusion CRM data inside Microsoft Outlook. The application is deployed to Outlook using the add-in framework and extends the Outlook data model and UI framework in order to store and render CRM data to the user.

How Oracle Fusion CRM Data is Displayed in CRM for Microsoft Outlook

Oracle Fusion CRM data is synchronized to users' computers and maintained in native Microsoft Outlook storage. While working in Outlook, users access CRM data that is stored locally, even when connected to the corporate network. The changes made to the CRM data are periodically synchronized with the Oracle Fusion CRM application. There are two options for storing the CRM data:

- A Microsoft Outlook mail profile configured to use a Microsoft Exchange service with the Use Cached Exchange Mode enabled to allow data to be stored in an offline storage file (.ost file format)
- A Microsoft Outlook mail profile configured to use the Internet E-Mail service with personal folder storage (.pst file format)

Because CRM data is maintained in Outlook storage, it can be displayed and accessed like any other Outlook item. For instance, CRM data types will appear in the folders for the user's mailbox alongside other native Outlook types, and users can select the CRM folder and view the CRM records there as they would work with other Outlook information. Within a given folder, the user can select and open a single record to view the data. In this case, the user will have access to CRM data that appears within an Outlook form or inspector window.

In addition to accessing CRM data in Outlook explorer views and inspector windows where the CRM data is the primary focus, users will also be able to access CRM context when viewing standard Outlook items like appointments, e-mails, and tasks. For these Outlook types, the user will be able to specify the CRM customer, related sales item, contacts, and resources associated with the Outlook item, and will be able navigate to the related CRM item to review additional details.

Data that is stored in either cached Exchange mode in .ost file format, or in personal folders in .pst format, is accessible to the CRM for Microsoft Outlook.
user while disconnected. The user interacts with the CRM data that is stored locally on his computer and periodically synchronizes data between Outlook and the Fusion CRM server. Synchronization happens when the user is connected to the corporate network and can access the CRM application server. Because the user always works with the local set of CRM data, he will have access to the data from the server immediately following the synchronization process, but doesn’t directly access or update the data on the server. Changes are made to the local data set, and then the synchronization process takes care of making changes to the local or server data sets to align the two.

Overview of the Synchronization Process

After CRM for Microsoft Outlook is installed, the user must perform an initial synchronization to retrieve his accessible CRM data. Several synchronization settings are configured as part of the First Run Assistant process that influence the initial synchronization. These include the frequency of automatic synchronization, the synchronization filters to use, and which objects are enabled or disabled from synchronization. These settings can be changed by the user after the initial synchronization. Once the user completes the First Run Assistant process, the initial synchronization will begin. The duration of the synchronization process will depend on the number of records that will be synchronized, network bandwidth, load on the server, as well as processing speed and memory available on the user’s computer. A rule of thumb is to try to configure synchronization filters so that no more than five to ten thousand records are synchronized.

During the synchronization process, the application performs the following steps:

1. Connects to the Fusion CRM server CRM for Microsoft Outlook synchronization services using SOAP over HTTP and authenticates the user.

2. Performs a check to determine the configuration for which the user possesses access. Access to an Outlook configuration is established based on a privilege associated with a user’s job role that allows access to an Outlook client deployment package.

3. If a user has access to a deployment package, it is downloaded, and the configuration is applied to the Outlook mailbox.

4. The final step is to synchronize data. The records that are retrieved depend on the internal filters configured on the server, data security applied to the objects that are synchronized, and the user filters.

Subsequent synchronization cycles follow a process that includes these steps:

1. CRM for Microsoft Outlook sends a request to the Fusion CRM server with a list of objects and the current user filters and requests a snapshot of IDs and timestamps for all records that are within the scope of the object list and specified filters.

2. The server sends a response with the requested information.

3. CRM for Microsoft Outlook makes a local snapshot of IDs and timestamps and compares that to the server snapshot.

The differences between the local snapshot of IDs and timestamps and the server snapshot result in a few possible actions:
• Inserts, updates, or deletes data on the Fusion server based on changes that occurred in CRM for Microsoft Outlook since the prior synchronization.

• Inserts, updates, or deletes data in CRM for Microsoft Outlook based on changes that occurred on the Fusion server since the prior synchronization.

In all cases, changes that are made to data locally in the CRM for Microsoft Outlook client are only sent to the Fusion server during the subsequent synchronization session; however, users who want to synchronize a change or set of changes immediately can start the synchronization cycle manually to avoid waiting for the next scheduled synchronization.

About Web Services Usage During Synchronization

The synchronization process on the Fusion server is supported by CRM for Microsoft Outlook accessing Web services. CRM for Microsoft Outlook accesses two Web services directly -- one that provides access to data during synchronization processing, and one that provides access to metadata. The synchronization process is initiated by CRM for Microsoft Outlook within the Outlook application, and the Fusion server accepts synchronization requests, routes them to the appropriate services within the service, and returns the appropriate responses. The work that each part of the synchronization architecture performs is summarized as:

1. CRM for Microsoft Outlook synchronization engine and connector that are deployed to Microsoft Outlook perform the following:

   • Initiates a new synchronization request based on a preconfigured automatic synchronization interval or by an ad hoc user request to start a new synchronization cycle.

   • Uses the stored details about username, password, server connection information, and CRM public security certificate stored on the user’s computer to format and send requests to the CRM application server.

   • Based on the configuration deployed to a user’s computer (including object types deployed), fields defined as part of those objects, synchronization filters and the like, the application generates the appropriate SOAP message content and expects the corresponding response when using the HTTP or HTTPS transport to communicate with the CRM application server.

2. The Fusion server hosts an application that listens for CRM for Microsoft Outlook synchronization requests, and the synchronization services perform the following:

   • The OutlookRequestHandlerService Web service processes all incoming requests for data synchronization, and the OutlookMetadataService Web service handles requests to retrieve metadata.

   • Incoming SOAP messages are routed to the appropriate service. These messages include one or more requests to invoke a method on the target service.
• Requests sent to the OutlookRequestHandlerService in particular are routed to other services to perform the action expected from the synchronization process. For instance, a request to get appointment data sent to the OutlookRequestHandlerService will be routed to the appointment Web service that will process the request and return the requested data, and the OutlookRequestHandlerService will send this back to the CRM for Microsoft Outlook client that sent the request.

A synchronization cycle will include requests to get a server snapshot, and can then include many additional requests to query, insert, update, and delete data based on the changes detected when CRM for Microsoft Outlook compares the local and server snapshots.

• Each of these requests is processed based on the type of request, and is either managed within the OutlookRequestHandlerService processing directly or is routed to the appropriate target service to be fulfilled.

Extensions to the Standard Outlook User Interface

In addition to standard Outlook data storage mechanisms and the synchronization engine, several extensions to the standard Outlook user interface provide a way to access and manage CRM data inside of Outlook. Examples of extensions to the standard Outlook user interface include custom toolbar buttons, menu items, inspectors that display Fusion CRM data, controls that are embedded on standard Outlook item inspectors, the personalization options dialog box, and so forth. The CRM for Microsoft Outlook client can use these extensions to perform a variety of tasks.

The following are some examples of tasks that the user can perform:

• Create, view, and edit CRM data in Outlook.

• Mark an Outlook item to be shared with CRM Desktop and associated sales data.

• Initiate a standard Outlook action, such as sending an e-mail or scheduling a meeting in the context of a sales item.

The behavior of the extended Outlook user interface is influenced by custom CRM business logic that performs a variety of validations during data entry. The following are some examples of validation that are performed:

• Confirm that the data type is valid for a given field.

• Make sure fields that are required are populated.

• Prevent changes to fields or records that are configured to be read-only.

• Validate field values based on comparisons with other fields or static values.

• Apply conditional validation so that a field may be required or read-only based on other criteria.

Physical Components that CRM for Microsoft Outlook Architecture Uses

Following are the major physical components that CRM for Microsoft Outlook uses:
1. CRM Database

This is the database accessed by the CRM application that stores data about customers, contacts, business opportunities, and so on.

2. CRM Application Server

This is the server that hosts the CRM for Microsoft Outlook application and the related Outlook Web services, and therefore is the main entry point for synchronization requests coming from the CRM for Microsoft Outlook add-in running on users' computers.

3. Laptop or Desktop

This is the computer where the CRM for Microsoft Outlook add-in is installed, and where users are working with CRM data in Outlook. The Outlook add-in will install binary files that support synchronization of CRM data and integration with Outlook, including support to extend the Outlook data model and user interface, and resource files containing images and strings to initialize the application. The CRM for Microsoft Outlook add-in will connect to the CRM application server and download the appropriate configuration and CRM data for the user which are also stored on this computer.

4. Corporate Messaging Infrastructure

The corporate messaging infrastructure encompasses all of the server computers and other network topology that support the transmission of e-mail messages, and other personal information management capabilities such as the corporate calendar, contact and task lists.

**CRM for Microsoft Outlook Functional Components**

Following are the CRM for Microsoft Outlook functional components:

1. CRM Extensions in Outlook

Extensions integrate with Outlook data storage and deliver additional business logic and extensions to the Outlook user interface to allow users to access and modify CRM data. CRM data is viewed with extensions to the Outlook user interface. Changes to CRM data are controlled by business logic and custom controls and then finally stored in Outlook data storage (for example, in a user's mailbox storage file). The user works with a version of the CRM application, as defined in the configuration deployed to the user's computer. Changes to CRM data since the last synchronization cycle are calculated by the synchronization engine during data synchronization with the CRM application server.

2. Synchronization Engine

The synchronization engine handles requests to initiate a synchronization cycle and is responsible for structuring the requests that are sent to the server. For the initial and incremental synchronization cycles, the synchronization engine manages requests to count records available to the user; sends a request to generate a server snapshot; initiates the process to generate a local snapshot; compares the results; and calculates the necessary requests to be sent to the CRM application server to complete the synchronization of local and server data sets. The synchronization
engine works in tandem with the connector to correctly format and transmit messages with the CRM application server.

3. CRM Connector

This part of the CRM for Microsoft Outlook add-in is responsible for knowing how to connect and communicate with the CRM application server. The connector uses details such as the username, password, connect string, public security certificate, and client metadata to interpret requests from the synchronization engine to correctly format and send requests to the CRM application server. All details of the requests to send to the server are orchestrated by the synchronization engine, but the transmission of the requests and retrieval of the responses is done by the connector. The connector uses the details in the connect string to know where to send requests to the CRM application Web services.

4. CRM Application Web Service

CRM Web Service provides functionality to handle the user session, and to add, delete, modify, count, and list data objects that are required by the Web service connector.

FAQs for Define Outlook Integration

What's a client configuration file?

In Oracle Fusion CRM for Microsoft Outlook, a client configuration file describes a part of the application configuration that resides on the user computer, and it extends the desktop application. Client configuration files can either describe a portion of the application logic implemented as Java script, or can be a declarative configuration of items, such as UI components or synchronization mappings implemented as XML. Each configuration file has a particular type. There can be more than one version of any file type at one time as long as the names differ, and only one file of any given type can be included in a deployment package.

What's a client deployment package?

In Oracle Fusion CRM for Microsoft Outlook, a client deployment package is a collection of metadata files that describe the CRM application extensions deployed to users' computers. Access to a given deployment package is given to CRM application users through a privilege associated with their job role. When a user connects to the CRM application server to synchronize data from a desktop application like Microsoft Outlook, the application determines if any changes to the package have occurred, and if so, downloads any changes.

What's a client configuration validation file?

In Oracle Fusion CRM for Microsoft Outlook, the client configuration validation file (.xsd) describes the structure of a valid client configuration file (.xml). The application uses the client configuration validation file to check that any client configuration file imported to the server is structured correctly and complies with the requirements of the validation file. The validation process happens automatically during the import of any client configuration file, and helps catch misconfigured files.
What's a server configuration file?

The Oracle Fusion CRM for Microsoft Outlook application uses a file to identify and map services and view objects that are used when processing synchronization requests, and to correctly query, insert, update, and delete data on the server. There is only ever one of these files used at a given time, and changes made to it are recognized by the application and loaded immediately.

Set Up Customer Center

Customer Center: Explained

Oracle Fusion Customer Center enables the comprehensive management of customer information. Customer Center collects data from various systems and presents them for management in one location.

Following are some of the capabilities of Customer Center:

- Create customers and contacts
- Update customers and contacts
- Maintain customer hierarchies
- Maintain competitor information

When working with Customer Center, be aware of the following terminology used throughout the application:

- Sales prospect
- Sales account
- Customer
- Consumer
- Legal entity
- Billing account

Sales Prospect

A sales prospect is a prospective sell-to entity, or person, at an existing or potential customer used to define Leads. A prospect is the lowest level representation of a business entity that your company's marketing processes will track and act upon. The sales prospect does not have a sell-to address. You can create a sales prospect from a party that does not have a sell-to address when you create the first lead for that party. You can also create sales prospects in Customer Center and by importing them in bulk.

You can create leads against sales prospects, but a sales prospect must be qualified and converted to a sales account before you can create opportunities for it. To qualify and convert a sales prospect, a set of business criteria or rules must be satisfied. For example, the prospect may be required to meet the criteria for account assignment.
**Sales Account**

A sales account is a specific sell-to entity within a given customer. You can create leads and opportunities against sales accounts. A single customer might have a collection of sales accounts. To avoid confusion when assigning territories to the account, each sales account has only one sell-to address. Typically, a sales team manages a sales account. The sales team is comprised of resources assigned to the territories associated with the sales account. Additionally, a profile option determines whether a sales account is a named sales account, an existing sales account, and the account owner. Named sales accounts are typically strategic accounts assigned to dedicated territories. An existing sales account is one where there is an existing financial relationship or had previous installs. You can create sales accounts in Customer Center and by importing them in bulk.

**Customer**

Within Customer Relations Management (CRM), sales accounts and sales prospects are collectively referred to as Customers. Additionally, a Customer also can have representations as a legal entity and a billing account that are expressed as root nodes in a hierarchy to the respective sales accounts for that customer.

View the Customer Hierarchy: A customer’s hierarchy represents a holistic view of the customer’s structure, showing you the customer type, the parent for the customer, the subsidiaries of the customer, as well as rolled up revenue analysis data.

**Consumer**

A consumer is a person who is a qualified paying individual. A consumer can have one or more sales accounts or sales prospects. A consumer is not an organization sales account or sales prospect. A consumer is by definition also a legal entity.

**Legal Entity**

A legal entity is a party that can enter into legal contracts or a business relationship, and be sued if it fails to meet contractual obligations. There are two types of legal entities: internal and external. A customer with a party usage of Legal Entity is considered an internal legal entity and is used for interdivisional selling within your own company. A customer with a party usage of External Legal Entity is any external customer who fits the definition of legal entity. Legal entities may also be used to group multiple sales accounts, sales prospects and other classes of entities or parties.

**Billing Account**

A billing account is a party that represents the financial account transactional entity for a given Customer.

**Customer Center Trees: Explained**

There are five types of Oracle Fusion Customer Center trees. Each tree displays slightly different nodes, and the information that you are able to view and edit on each node depends upon your security privileges and your membership status on the sales account team.
The five types of Customer Center trees are:

- Customer
- Consumer
- Contact
- Legal Entity
- Account Plan

**Customer Tree**

The customer tree displays nodes for business-to-business entities such as sales account and sales prospect. If you are a member of the sales account team with at least Edit level access or you have the Sales Party Administration duty, you can update information on the following nodes: contacts, organization chart, classifications, assessments, discussion forums, classifications, assessments, notes and, account assessments. Only those users with Sales Party Administration duty or Full level access on the sales account team and profile nodes can update the members of the sales account team.

**Consumer Tree**

The consumer tree displays nodes for business-to-consumer entities such as sales accounts and sales prospects. All nodes on the consumer tree are visible to all users. Since consumers are individuals rather than organizations, there are fewer nodes displayed in the consumer tree than in the customer tree. For example, you will not see nodes for organization chart, contacts, assessments, or others that are pertinent to organizations.

**Contact Tree**

The contact tree displays nodes for contacts with the contact profile and other related information such as the customer (the organization) to whom the contact belongs, interactions with the contact, notes, and so on. All nodes on the contract tree are visible to all users.

**Legal Entity Tree**

The legal entity tree displays nodes for business-to-business entities that are marked with a party usage of Legal Entity. Because legal entities are typically a parent or root node in a customer hierarchy, legal entity tree has a subsidiaries node. All nodes on the legal entity tree are visible to all users.

**Account Plan Tree**

An account plan is a plan to sell to a certain set of accounts in a coordinated way. There is a specific set of assigned resources. This tree shows associated contacts, assets, Opportunities and member accounts of the plan.

**Sales Account Team Member Access Level: Explained**

There are three types of sales account team memberships known as access levels. These access levels control the team member’s privileges for the sales account:

- View Only
- Edit
• Full

When a resource is initially added to the sales account team, a profile option setting determines the member’s default access level. If that member is removed from the sales account resource team, she no longer has access to the sales account, unless she is still a member of a territory that is assigned to the sales account. Resources in the management hierarchy of a newly added team member inherit the same access level of the subordinates.

**View Only**

View Only is the minimum level assigned to a sales account team member. This access level enables the team member to view the contents of the sales account child attributes such as sales account team, snapshot, assessments, discussion forums, notes, interactions, appointments, and tasks. This assumes, however, that the team member also has functional access to view that child attribute. If the team member's resource role does not provide functional access to view a particular child attribute of a sales account, that member cannot view the attribute, regardless of her sales account team access level. A team member with View Only access level for a sales account can view only the opportunities, leads, and revenue lines to which she has relevant data privileges.

**Edit**

Sales account team members with the Edit access level can view and edit all customer-related objects. They can view and edit only the opportunities, leads, and revenue lines to which they have the relevant data privileges. The Edit access level provides a sales account team member with the ability to run the territory reassignment process, but she cannot change the composition of the sales account resource team.

**Full**

The Full access level allows team members to do everything that the Edit access level allows, with the addition of being able to change the composition of the sales account resource team. A team member with Full access can manually add and remove team members, change a member’s access level, and mark the lock assignment setting for team members. When a sales account is created, only the sales account owner and sales administrators are granted the Full access level, but they can grant Full access to other team members.

### Sales Account Territory Member Access: Explained

Access for the Territory owners and members parallels that of the Sales Team members.

These access levels control the internal and partner territories privileges for the sales account:

- Internal territory owner: Full access
- Internal territory members (non-owner): Edit access
- Partner territory owner and members: View-only access

**Note**

Territory Management must be implemented to utilize this feature.
How Assignment Object Components Work Together

The work object, candidate object, and attributes are components that fit together to create assignment objects that are used in rule-based and territory-based assignment. Work objects are business objects that require assignment such as leads and opportunities. Candidate objects are business objects such as resources and territories that are assigned to work objects.

When you create candidate objects, you can select attributes for them that are later used in rules or mappings. These candidate objects also become candidates that are available for association when you create work objects. When you create work objects, you can select attributes for them also, as well as associating one or more candidates.
Work Object

A work object is a business object that requires assignment such as a lead or an opportunity. Creating a work object involves entering its application information, selecting its attributes to use during assignment, and associating one or more candidates.

Candidate Object

A candidate object is a business object such as a resource or a territory that is associated with one or more work objects for eventual assignment. Creating a candidate object involves entering its application information and selecting its attributes to use in rules or mappings. A special type of candidate object is a classification object. This type of candidate object does not represent a business object that gets assigned to a work object. It is used only with classification rules and is used primarily to rank or qualify leads.

Note

As candidate objects are created, they become available as candidates that can be associated with one or more work objects as part of the work object creation process.

Attributes

Attributes are elements in the view object defined for an assignment object. For each assignment object, you can select one or more attributes that you want to use when configuring assignment rules or mappings. For example, for a work object like sales account, you might choose the attributes of Named Account Flag, Customer Size, and Organization Type. When you configure assignment rules for the sales account work object, your chosen attributes are available for your rule conditions. In other words, you could configure a rule for sales account using the Named Account Flag attribute, and set a condition where the assignment engine looks for sales accounts that have their Named Account Flag equal to Yes.

When selecting attributes for a candidate object, you will not only select the attributes you want to use when configuring assignment rules and mappings that involve that candidate object, but you also want to select the attributes for that candidate object that you want to appear in the screen that displays recommended candidates after assignment manager is run. For example, if a candidate object is resource (sales representative), and you want to show sales representatives' first names, last names, and phone numbers when they are recommended during assignment processing, you need to select the attributes for the resource candidate object that correspond to first name, last name, and phone number, and you need to specify the order in which these attributes will appear in the recommended candidates screen.

Note

This feature is not used by any CRM applications at this time.

Adding an Assignment Object

The Manage Assignment Objects pages enable you to define and edit the Work and Candidate objects as well as define any territory-based mappings. The figure
above shows the relationship between the work and candidate objects and the mapping of the matching candidates to work objects.

When you add or edit a work or candidate object there are several key pieces of information that are required in the definition:

• Name: a unique name for the object with an optional description.

• Code: a unique code used in processing the object.

• Work/Candidate Object check boxes: indicates if the object is a work object, candidate object or both.

• Application Module: an Oracle Application Development Framework (ADF) business component that encapsulates the business service methods and UI-aware data model for a logical unit of work related to an end-user task. Enter the fully qualified definition name of the consumer application, Application Module. Valid for top level Work and Candidate objects. Child objects automatically inherit this value from its parent.

• Application Module configuration: Valid for Top Level Work and Candidate objects except Classification Candidate objects. Child objects will automatically inherit this value from its parent.

• View Object Instance: used to define the data model of a view object component when designing an application module, for example, Opportunity. Valid for all levels of Work and Candidate objects except Classification Candidate objects.

• View Criteria may be defined to filter the information for the rows of a view object collection. Valid for top level Work and Candidate objects except Classification Candidate objects.

• Primary Key Attribute 1: First or only attribute that makes up the object primary key. Valid for top level Work and Candidate objects except Classification Candidate objects.

• Refresh Interval: the number of minutes between refreshes of candidate object data. The default setting is 0 minutes. Valid for top level Candidate objects except Classification Candidate objects.

• Initial Caches: The initial size of the cache when processing an object. This value will be used the first time the engine processes objects or following a server bounce. The default value is 2, and the maximum value is 20. Only valid for top level Candidate objects except Classification Candidate objects.

Note

All Work Objects that are used for scoring, Lead for example, use the Product Level (MOW_SCORING_INITIAL_CACHES) Initial caches for scoring rules profile option value.

• Maximum Caches: The maximum size of the pool/cache when processing the object. The default value is 5, and the maximum value is 25. Only valid for top level Candidate objects.
All Work Objects that are used for scoring, Lead for example, use the Product Level (MOW_SCORING_MAX_CACHES) Maximum caches for scoring rules profile option value.

- Score Attribute: The attribute on the object that stores the total calculated score after an assignment request has been processed. Valid for top level Work objects only.
- Assignment Date Attribute: The attribute on the object that stores the assignment date after an assignment request has been processed. Valid for top level Work objects.
- Exclude Assignment Attribute: The attribute on the object that stores the setting for excluding a work object from assignment. Valid for top level Work objects.

Adding Assignment Object Attributes

Assignment Manager allows users to specify a set of attributes from the assignment object VO to be used during the assignment evaluation. The assignment engine will load these Assignment Object Attributes for each assignment object VO row, in addition to any primary key or assignment attributes. This is designed to improve performance by not loading those attributes not used for the assignment evaluation.

Assignment Object Attributes should be defined for each work object and any child objects as well as each candidate object to be used by the Assignment Engine.

- View Object Attribute: Name of each attribute in the view object defined for the assignment object. Assignment Rules or Mappings can be configured using these attributes. For Candidates Objects, the attributes that appear in the interactive assignment UI should also be selected.
- Candidate Information Sequence: The sequence that this attribute is displayed in the Interactive Assignment UI.

Defining Work Object and Candidate Object Associations

The administrator needs to define the association between the work object and candidate object. For example the Lead work object may have an association with both the Territory candidate object and the Resource candidate object. This implies that Assignment Manager can be used to assign Territories and Resources to a lead.

Configuring Assignment Manager: Critical Choices

Assignment is the process for selecting a candidate as an object and executing the association to a work object. Assignment consists of two phases. The first phase is the matching phase, where matching rules or mappings are evaluated to find the right assignees from a list of possible candidates. The second phase is the disposition phase, where the disposition, or assignment, of matching candidates is handled. Assignment Manager is the tool used to establish the business objects that require assignment, to set up the resources that can be assigned, and to create the rules and mappings that dictate the selection and assignment of those resources. Candidates are potential assignees for a work
A work object is a representation of an application business object inside Assignment Manager. A work object captures the attributes of a business object and associated child objects to be used for matching purpose. To best plan the configuration of Assignment Manager, you should consider the following points:

- **Business objects**
- **Resources**
- **Assignment disposition**
- **Attributes**
- **Mappings and rules**

**Business Objects**

A business object is a data entity or a collection of data treated as a unit, such as a sales account, an opportunity, or a lead. Any business object that requires the assignment of a resource to act upon it is considered a work object by Assignment Manager. The work object is a representation of the business object, and mappings and rules are developed to ensure timely and accurate assignment of candidates (for example, territories or resources) to those work objects. When configuring Assignment Manager, carefully consider which of your business objects require assignment, and create work objects only for those that do.

**Assignment Disposition**

After you determine the business objects (work objects) that require assignment and the candidate objects that you will assign to them, you must decide how the matching candidate assignment disposition will be carried out. Consider these questions:

- Do you want to assign a single resource or multiple resources?
- Do you want to automatically assign matching candidates or run custom logic against matching candidates?
- Do you want to record the matching candidate score on the work object?
- Do you want to retain manually assigned candidates when assignments are processed?
- Do you want to replace disqualified candidates when assignments are processed?

**Attributes**

To ensure that candidates are properly assigned to work objects, you will create mappings and rules. These mappings and rules employ attributes to determine the best assignments. As you set up work objects and candidate objects in Assignment Manager, you will also select the attributes of those objects that you want to use in your mappings and rules. For example, you might want to assign a resource such as a sales representative to a business object like opportunity based on the product skill of the sales representative. In this case, when you create the opportunity work object and the sales representative candidate object, you will select the attributes of opportunity and sales representative that correspond with product skill. Selecting these attributes makes them available for mappings and for conditions on your rules, so ensure that you select the
attributes that reflect the criteria that you want to use for matching business objects to work objects.

**Mappings and Rules**

Assignment mappings drives territory-based assignment. These mappings identify the dimensions, attributes, and territory filtering used in territory-based assignment processing. A default set of mappings are seeded. This seeding assumes that opportunities, leads, and sales accounts use the same territory hierarchy. Rules are defined for the execution of rule-based assignment. Rules are designed to return candidates based on whether these candidates match a set of criteria, are within a defined scoring range, or are of a specific classification.

You create the mappings and rules using the work objects, candidate objects, and attributes that you already established. When designing your mappings and rules, carefully consider how you want to match candidates to work objects. For example, would you want resources assigned based on their geographic location, or their product knowledge, or their skill level, or a combination of any of these attributes? Do you want to match candidates only, or would you like to match candidates and score them? In a multiple-candidate scenario, do you want to assign all matching candidates or only those who achieve higher than a specific score? Consider these questions before creating mappings and rules.

**Sales Account Assignment Object: Explained**

Territory-based assignment is based on intelligent mapping of sales account assignment object attributes and sales territory dimensions. The Sales Account Assignment object is used by Assignment Manager to identify the sales accounts and then determine which territories to assign. The table below lists sales account assignment object attributes and corresponding customer attributes as shown in Customer Center Profile and Classification nodes. See Configuring Assignment Manager: Critical Choices for more information about the assignment process.

<table>
<thead>
<tr>
<th>Sales Account Assignment Object Attribute</th>
<th>Corresponding Customer Center Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography ID</td>
<td>Sell-to Address</td>
</tr>
<tr>
<td>Industry</td>
<td>Primary Industry: the primary classification code for the classification category defined in profile option Industry Classification Category.</td>
</tr>
<tr>
<td>Organization Type</td>
<td>Primary Organization Type: the primary classification code for the classification category Organization Type defined in profile option Industry Classification Category.</td>
</tr>
<tr>
<td>Customer Size</td>
<td>Customer Size</td>
</tr>
<tr>
<td>Named Account Type</td>
<td>Named Sales Account Indicator</td>
</tr>
<tr>
<td>Party ID</td>
<td>Party ID</td>
</tr>
<tr>
<td>Auxiliary Dimension 1</td>
<td>the primary classification code for the classification category defined in profile option Industry Classification Category for Auxiliary Dimension 1.</td>
</tr>
<tr>
<td>Auxiliary Dimension 2</td>
<td>the primary classification code for the classification category defined in profile option Industry Classification Category for Auxiliary Dimension 2.</td>
</tr>
</tbody>
</table>
Scheduling Sales Account Assignment: Explained

The Sales Account assignments process can be scheduled and run on the Scheduled Process page. You need to have the 'Run Sales Party Batch Assignment' privilege to be able to define and run sales account batch assignment.

To access the Scheduled Process page, start on the Fusion Home page and click Navigator. Under the Tools heading, click Scheduled Processes.

1. Click Schedule New Process then click type Job. Choose the process named SalesAccountBatchAssignRequest. If needed, use the Search link at the bottom of the Search window.

2. Enter your process details. The following table shows the view criteria and its description, as well as any bind values that are required.

   - **Work Object code**: Sales_Account_Work_Object
   - **Candidate Object Code**: SalesAccountTerritory_Candidate_Object
   - **Assignment Mode**: Territory
   - **View Criteria Name**: (see table below)
   - **View Criteria Bind Values**: (see table below)

<table>
<thead>
<tr>
<th>View Criteria Name</th>
<th>View Criteria Description</th>
<th>View Criteria Bind Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SalesAccountsUpdatedSinceVC</td>
<td>Use this view criteria to assign sales accounts which have not been previously assigned and have LAST_UPDATE_DATE (in the ZCA_SALES_ACCOUNTS table) greater than the specified date. For newly created sales accounts, LAST_UPDATE_DATE is the same as the creation date.</td>
<td>BindLastUpdateDate=[YYYY-MM-DD HH:MM:SS]</td>
</tr>
<tr>
<td>SalesAccountsAssignedBeforeVC</td>
<td>Use this view criteria to reassign sales accounts which have been previously assigned and have LAST_ASSIGNED_DATE (in the ZCA_SALES_ACCOUNTS table) less than the specified date.</td>
<td>BindLastAssignedDate=[YYYY-MM-DD]</td>
</tr>
</tbody>
</table>
SalesAccountTerritoryBatchVC
Use this view criteria to reassign sales accounts impacted by the specified territory and territory dimensional realignment batch.

This view criteria is also used internally to invoke immediate/automatic assignments after territory proposal activation and territory dimension updates.

BindReassignmentBatchId=[Territory Reassignment Batch ID]

SalesAccountBulkImportVC
Use this view criteria to assign sales accounts created in a given customer import batch.

This view criteria is also used internally to invoke immediate/automatic assignments after customer import.

BindReassignmentBatchId=[Import Activity ID]

SalesAccountDimsForPartyVC
Use this view criteria to assign the sales account with the specified sales account ID.

BindPartyId=[Sales Account ID]

3. Define a schedule as needed using the Advanced button on the Process Details page. You can schedule the process to run as soon as possible, or to run at a given frequency and start date.

4. Submit your job and monitor it using the Scheduled Processes list, refreshing it to view the latest status updates.

Creating Assignment Mappings: Examples

For territory-based assignment, you create work-object-to-candidate-object mappings during assignment object creation. These mappings are used to make candidate assignments. You can create multiple types of mappings for assignments. The following scenarios illustrate these different mappings:

- Creating an attribute mapping
- Creating a dimension mapping
- Creating a literal mapping

Creating an Attribute Mapping

You want to assign territories to a sales lead when the territory program ID is the same as the sales lead program ID. Create a mapping where the work object is sales lead and the candidate object is sales lead territory. Select the territory
when the attribute territory program ID is equal to the sales lead attribute program ID.

Creating a Dimension Mapping

You want to assign territories to opportunity revenue lines based on the product associated with the revenue line. Create a mapping where the work object is opportunity revenue line, and the candidate object is territory. Select the product dimension as the mapping type. The candidate object low and high attributes correspond to the names of the low sequence and high sequence attributes for product on the territory. The work object low and high attributes correspond to the names of the low sequence and high sequence attributes for product on the revenue line. For example, the low sequence attribute for product on the revenue line might be called ProdSeqLow.

• Mapping using alternative attributes: Using the same scenario of assigning territories to opportunity revenue lines based on the product associated with the revenue line, you might encounter a situation where a revenue line does not have a product assigned to it, but it does have a product group assigned to it. Create the same mapping that you created for the dimension mapping scenario, and add the names of the low sequence and high sequence attributes for product group for the work object alternate low and high attributes. For example, the alternate low sequence attribute for product group on the revenue line might be called ProdGrpSeqLow.

• Mapping using default values: Using the same scenario of assigning territories to opportunity revenue lines based on the product associated with the revenue line, you might encounter a situation where the low sequence and high sequence attributes for product on a revenue line do not contain values when assignments are processed. Create the same mapping that you created for the dimension mapping scenario, and add low and high default values for the product attribute for revenue lines.

Creating a Literal Mapping

Literal mappings are a way of filtering the matched territories based on specific values of a territory attribute. You want to find only territories that are finalized (for example, territory status equals FINALIZED).

Turning a Business Process into a Task Template: Example

This example illustrates how to create a task template that represents a business process.

Scenario

A sales manager wants to create a task template for her department's client product demonstration process.

Client Product Demonstration Activities

The client product demonstration process occurs regularly. The sales manager does not want to manually create tasks for this process every time it occurs, so she decides to create a task template that includes the business process activities. Each time she repeats the business process, she can use the task template to automatically generate the appropriate tasks that need to be performed.
Analysis

The business process consists of the following activities:

- Book a conference room.
- Create an agenda.
- Confirm the date and time with the client.
- Make arrangements with presenters.
- Deliver product demonstration.
- Follow up with client.

Based on the analysis of the business process, the following task template is created:

Task Template Name: Client Product Demonstration

<table>
<thead>
<tr>
<th>Task</th>
<th>Category</th>
<th>Lead Days</th>
<th>Duration Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book conference room</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Create agenda</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Confirm date and time</td>
<td>Call</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>with client</td>
<td>Call</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Schedule presenters</td>
<td>Preparation</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Deliver demonstration</td>
<td>Demonstration</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Follow up with client</td>
<td>Call</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

Defining Notes: Points to Consider

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:

- Note Types
- Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn’t exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without
Define Common CRM Configuration

3-89

a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article Extending CRM Applications: how it works.

Assessment Templates: Points to Consider

Assessment templates let you analyze the health of a business object, such as a lead or an opportunity, and suggest appropriate next steps based on its diagnosis. To best plan and create assessment templates, you should consider the following points:

- Ratings
- Questions, Question Groups, and Question Weights
- Responses and Scores
- Associated Task Templates

Ratings

A rating is a textual qualification such as Excellent. There are three delivered ratings in the assessment template: Excellent, Average, and Poor. Ratings provide a metric other than a numerical score for qualifying the outcome of an assessment. Ratings are created at the beginning of the assessment template creation process. They are later applied to possible responses to questions in the template, which associates each rating with a score. An appropriate feedback will be displayed to you based on the completed assessment score once you submit an assessment. When setting up ratings and applying them to possible responses, it is important to remember that they and their associated feedback text will eventually display as part of the overall assessed health of a business object.

Questions, Question Groups, and Question Weights

Questions are the main components of an assessment template. They are written such that they aid in systematically determining the health of a business object, and they are grouped into logical collections called Question Groups. Each question in the template is assigned a question weight, expressed as a percentage, which is the relative importance of the question within the template. When an assessment template is used to perform an assessment, a question’s weight is multiplied by the response score given for the question to produce a weighted score for that question. When setting up questions, question groups, and question weights, it is important to carefully analyze which
factors determine the health of a particular business object (like a lead or an
opportunity) in your organization. Use those factors to create your question
groups; and then, for example, write three to five questions per group that are
weighted according to your analysis. There is no limit to the number of questions
that can be in a question group, but each question group must have at least one
question.

**Responses and Scores**

Responses are attached to questions in the template. Each question should have
at least two responses, unless it’s a free-form only question. More than one
response can be tied to the same rating but, between all of its responses, each
question should accommodate at least two ratings, unless it’s a free-form only
question. For example, if your ratings are Excellent, Average, or Poor you may,
for each question, include two responses that correspond to at least one of those
ratings, such as average. There must be enough responses to cover at least two
of the ratings such as Excellent and Average. You assign a score to each response
for a question, and the application normalizes the score based on a standard
scoring scale. When an assessment template is used to perform an assessment,
a question’s weight is multiplied by the normalized score of the response
given for the question to produce a weighted score for that response. When
adding responses to questions, ensure that the scores and ratings you assign to
each response correlate. In other words, the higher the score you assign to the
response, the higher the rating should be so that you have a strong quantitative
relationship between the two. Also note that you can allow free-form responses
for one or more questions in the template, but free-form responses are never
scored.

**Associated Task Templates**

A task template is an instruction to generate a group of related activities. You
can associate task templates with an assessment template in order to recommend
tasks that should be performed after an assessment has been done for a business
object. When you associate task templates with an assessment template, you
can indicate a score range for each task template, and based on the total score
of any assessment that uses your template, one or more task templates will
be recommended as follow-up activities. In order for a task template to be
available to associate with an assessment template, it must be assigned to the
same business object type as that assigned to the assessment template, and it
must have a subtype of Assessment. Ensure that you have set up task templates
correctly before attempting to associate them to assessment templates.

**Assessment Template Status Codes: Explained**

Throughout the life of an assessment template, it can be assigned several
different status codes.

These status codes control the actions you are allowed to make against an
assessment template.

- In Progress
- Active
- Retired
In Progress

This is the initial status of an assessment template. When an assessment template is at this status, you can edit any part of it. This is the only status at which you can delete a template. If the template is not deleted, it moves to the Active status next.

Active

This is the status assigned when the assessment template has been deployed for general usage. When an assessment template is at this status, you can make only minor textual edits to it, including, but not limited to, template description, question text correction, question sequencing change, response description, and score range feedback. From this status, you can move the template to Retired; you cannot delete it.

Retired

When an assessment template is at this status, it is no longer available for general usage. You cannot edit any part of it, and you cannot move it to any other status; however, it can still be copied. Active templates that are deleted revert to this status.

Assessment Template Score Range: How It's Calculated

The application calculates the score range for an assessment template using the question weights and the ratings and scores assigned to the possible responses for all the questions in the template. This topic explains when the score range is calculated and the components that are used in the calculation, so that you can make the best decision regarding the feedback text to apply to each score range. In addition to the automatic score range calculation, a manual method for adjusting score range is also available on the administration UI.

Settings That Affect Score Range

In order for the application to calculate the assessment template score range, you must:

- Apply weights to all template questions.
- Configure ratings and apply them to possible responses for all template questions.
- Apply a score to each of the possible responses for all template questions.

How Score Range Is Calculated

The score ranges for each rating in an assessment template are determined using the lowest and the highest weighted response scores for each question. So for each rating score range, the lower end of the range starts where the previous rating range ended, and the higher end of the range is the sum of the highest weighted scores that can be attained for that rating.

This table displays a simple example of the components used in the score range calculation.
<table>
<thead>
<tr>
<th>Question (Weight)</th>
<th>Response (Normalized Score)</th>
<th>Weighted Score</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the customer win? (20%)</td>
<td>Lower Operating Cost (100)</td>
<td>20</td>
<td>Excellent</td>
</tr>
<tr>
<td></td>
<td>Higher Revenues (80)</td>
<td>16</td>
<td>Average</td>
</tr>
<tr>
<td></td>
<td>Other (53)</td>
<td>11</td>
<td>Average</td>
</tr>
<tr>
<td></td>
<td>Don't Know (27)</td>
<td>5</td>
<td>Poor</td>
</tr>
<tr>
<td>What is our win? (80%)</td>
<td>Reference (60)</td>
<td>48</td>
<td>Average</td>
</tr>
<tr>
<td></td>
<td>Resale (50)</td>
<td>40</td>
<td>Poor</td>
</tr>
<tr>
<td></td>
<td>Partnership (100)</td>
<td>80</td>
<td>Excellent</td>
</tr>
</tbody>
</table>

This table displays the score range calculation based on the components from the first table.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Score Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excellent</td>
<td>65 - 100</td>
</tr>
<tr>
<td>Average</td>
<td>46 - 64</td>
</tr>
<tr>
<td>Poor</td>
<td>0 - 45</td>
</tr>
</tbody>
</table>

**Note**

If a template administrator does not use a particular rating while assigning ratings to possible responses, this could result in improper score range calculations. To counteract this problem, the score range calculation uses a built-in correction algorithm to ensure proper score ranges. The correction algorithm works like this: For a question where a particular rating is skipped, the low score for the skipped rating is calculated to be equal to the high score of the next lower ranked rating. The high score for the skipped rating is calculated to be equal to the low score of the next higher ranked rating.

Using the ratings displayed in the tables above, if the rating Average is not used for a question’s possible responses, the score range calculation assigns a low score to Average for that question that is equal to the high score of Poor for that question. It also assigns a high score to Average for that question that is equal to the low score of Excellent for that question. This ensures that the overall template score range for Average is calculated to fall between the score ranges for Poor and Excellent.

**Assessment Template Components: How They Fit Together**

The question weight, response score, and response rating are the assessment template components that fit together to calculate and display the overall assessment score, rating, and feedback text.

A question weight is multiplied by a response score to achieve a weighted score for an assessment template response. The weighted scores for all responses are added together to determine the total assessment score. This score will fall within a precalculated score range that is associated with a response rating and feedback text. Therefore, the score range within which the total assessment score falls determines the rating and feedback text to display for a completed assessment.
Question Weight

The question weight is the relative importance of a question within an assessment template, and it is expressed as a percentage. All of the question weights within a template must total to exactly 100. When an assessment template is used to perform an assessment, a question’s weight is multiplied by the score of the response given for the question to produce a weighted score for that response.

Response Score

A response score is the score assigned to a possible response to a question in the template. The template administrator sets response scores with no upper or lower bounds, and each score is normalized in order to accurately score an assessment that uses the template. The response scores are normalized by assigning a score of 100 to the highest response score, and then all other responses are assigned a normalized score relative to that highest score.

When an assessment template is used to perform an assessment, the normalized score of the response given for the question is multiplied by the question’s weight to produce a weighted score for that response.
Response Rating

A response rating is the rating assigned to a possible response to a question in the template. A rating is a textual qualification like Excellent or Poor that provides a metric other than a numerical score for qualifying the outcome of an assessment. A response rating is directly related to a response score, and this relationship should ensure that a higher score will translate to a higher rating.

Early in the template creation process, the administrator configures ratings to assign to responses. The administrator then assigns scores and ratings to responses, and the system calculates score ranges based on those entries. Each rating is assigned to a score range, and the administrator is given the opportunity to apply feedback text to the rating-score range combination.

When an assessment template is used to perform an assessment, the weighted scores from all responses are added to determine the total assessment score. That score will fall somewhere within the calculated score ranges, which then determines which rating is assigned to the assessment and what feedback text to display. The maximum total assessment score is 100.

Assessment Templates and Task Templates: How They Fit Together

One of the steps for creating an assessment template is associating task templates. You would take this step if you want to recommend sets of tasks to be done after an assessment is performed using your template. You associate task templates to ranges of scores in the assessment template, and where the overall assessment score falls within those ranges determines the tasks that are suggested to be performed after the assessment.
Assessment Template

An assessment template is a set of weighted questions and possible responses used to evaluate the health of a business object such as an opportunity or a lead. An assessment template can be associated with one or more task templates that are recommended based on the outcome of an assessment.

Task Template

A task template is an instruction to generate a group of related activities. By marking a task template with a subtype of Assessment, you make that task template available for association with assessment templates. The task template's business object type should be the same as that assigned to the assessment template. When an assessment is performed using an assessment template that has associated task templates, one or more task templates are recommended based on the total score of that assessment and can be used to generate a list of activities to perform.

For example, you can associate a task template called Engage Business Development Manager with your assessment template called Potential for Win-Win. Associate the task template with the score range of 86 to 100, so if an assessment using the assessment template Potential for Win-Win scores within that range, the application recommends the Engage Business Development Manager task template and a list of follow-up activities based on that template can be generated.

Customizing Oracle Fusion Customer Center Pages Using Oracle Composer: Explained

In Oracle Fusion Customer Center, you can customize the Sales Account region on the customer profile page using Oracle Composer. To access Oracle Composer, navigate to the customer profile page and select to customize the page from the Administration menu in the global area. You can also access Oracle Composer by selecting Customization Manager from the Administration menu.

Available Customization Options

When you select to customize a page from the Administration menu in the global area, you launch Oracle Composer.

The customizations that you make to the customer profile are applied based on your layer selection:

- **Site**
  
  Your customizations are visible to all users.

- **External or Internal**
  
  Depending on your selection, your customizations are visible to either external or internal users.

  External users could be your partners or anonymous users. Internal users could be your employees.
• Job Role

Your customizations are visible to users who have the selected job role.

If a user has more than one job role, then the sequence in which customizations are applied is alphabetically ordered by job role name.

Oracle Composer provides two views for working with page content: Design View and Source View. Design View provides a WYSIWYG rendering of the page and its content, where controls are directly selectable on each component. Source View provides a combined WYSIWYG and hierarchical rendering of page components, where controls are available on the header of the hierarchical list. Source View provides access to page layout components that are otherwise not exposed on the page, and therefore not available in Design View.

This table lists the types of customizations available for the customer profile page. You can perform most of the basic customizations in either Design View or Source View. Some customizations, however, must be completed only in Source View.

<table>
<thead>
<tr>
<th>Oracle Fusion Customer Center Page</th>
<th>Customization Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer profile</td>
<td>Show and hide components on a page:</td>
</tr>
<tr>
<td></td>
<td>• Show and hide fields within the Sales Account region by selecting the Show Component check box on the Display Options tab in the Component Properties dialog for a component.</td>
</tr>
<tr>
<td></td>
<td>• Show and hide the Sales Account region itself by selecting the Show Component check box on the Display Options tab in the Component Properties dialog for the container.</td>
</tr>
<tr>
<td></td>
<td><strong>Tip</strong></td>
</tr>
<tr>
<td></td>
<td>You might have to scroll down to see the Show Component check box.</td>
</tr>
<tr>
<td>Customer profile</td>
<td>Move components on a page:</td>
</tr>
<tr>
<td></td>
<td>• Move components within the Sales Account region (only) by cutting and pasting them in Source View, or by rearranging them on the Child Components tab in the Component Properties dialog for the container.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>The components available for updating include any custom fields that were added to the Sales Account object using the CRM Application Composer.</td>
</tr>
<tr>
<td>Customer profile</td>
<td>Make an updateable field read only by selecting the Read Only check box on the Display Options tab in the Component Properties dialog.</td>
</tr>
</tbody>
</table>
**Define Common CRM Configuration**

<table>
<thead>
<tr>
<th>Customer profile</th>
<th>Make a field required by selecting the <strong>Show Required</strong> check box on the <strong>Display Options</strong> tab in the Component Properties dialog.</th>
</tr>
</thead>
</table>

**Customization Privileges Required**

The customizations that you can make using the Oracle Composer Source View are available to you only if your assigned job role includes the **Page Composer Source View Access Duty** duty role. Contact your security administrator for details.

**Note**

Your assigned job role must also include the **Administration Link View Duty** duty role to access Oracle Composer at all, even if only the Design View. This duty role exposes the Administration menu in the global area.

**FAQs for Set Up Customer Center**

**How can I personalize the customer center tree?**

Personalizing the Oracle Fusion Customer Center tree enables you to have a more intuitive navigation experience. The tree, located in the regional area of the page, is made up of object nodes such as Profile or Contacts. To personalize the tree, use the Action menu located directly above the tree or right click on any tree node, and click Manage Customer Tree in the menu popup. The Manage Customer Tree window will pop up. Select the node you wish to modify. You can change the name, whether the node is visible or not and if it should the default node that will display upon opening the tree. When you save, the customization will be associated to your user name.

**How can I create a task template that is available to associate with assessment templates?**

Create the task template with a subtype of Assessment.

**What happens if I mark an assignment object or one of its attributes as inactive?**

When the assignment object inactive box is checked the selected work or candidate assignment object is not available for assignment processing. When the assignment attribute inactive box is checked the selected work or candidate object attribute is not available for assignment processing.

**Note**

The object or attribute cannot be set to inactive if there is a mapping or rule defined using the object or attribute.

**What’s the difference between literal, dimension, and attribute mapping?**

Attribute Mapping: This mapping enables you to compare and match attribute values between a work object attribute and a candidate attribute. When
the candidate attribute matches the work object attribute the candidate is selected. Attribute mappings should be used when the work object and candidate object attributes in the comparison are non-dimensional attributes. For example, consider a lead work object with a program ID attribute and the territory object with program ID attribute. The selection criteria is: select Sales Territories where Sales Territory.TerritoryLocation equals Sales Opportunity.OpportunityLocation The assignment engine will use this mapping data to construct a query on the candidate object that is equivalent to the selection criteria. When creating the mapping, use the Function Code field to specify a unique identifier for the dimension. This identifier will be passed to the translation function, in case the same function is used for multiple dimensions.

Literal Mapping: Literal Mapping is used almost exclusively to filter the candidate objects. This form of mapping enables the comparison of candidate attributes against a specific value chosen by the user. The assignment engine will compare the mapped candidate object attribute against the specified literal value. For example: Select the Territory Candidate object that has the attribute TerrStatusCode that equals the value FINALIZED.

Dimension Mapping: Dimension mapping should be used when the work object and candidate object attributes in the comparison are dimension attributes, such as Geography, Product, or Account. When creating the mapping, use the Function Code field to specify a unique identifier for the dimension. This identifier will be passed to the translation function, in case the same function is used for multiple dimensions.

When do territories get assigned to sales accounts?

Internal territories get assigned to sales accounts in the following scenarios.

- When sales accounts are created.
- When a sell-to address is added to an existing sales party.
- When sales accounts are imported in bulk.
- When certain attributes on sales accounts that correspond with territory assignment dimensions are updated.
- When batch assignment is run.
- When you select the Assign Territories menu action on the Sales Account Team node for the sales account
- When territories are realigned or when personnel leave the territory or the company.

Note

The following profile options determine whether territory assignment and reassignment is automatic for sales accounts. The default setting for both is YES.
• Sales Account Automatic Assignment on Create Enabled
• Sales Account Automatic Assignment on Update Enabled

Automatic assignments are always enabled following an import, party merge and territory realignment.

During initial implementation and migration, it is possible to create sales accounts before territories have been set up in the system. These sales accounts will not receive any territory assignment because there are no territories. These accounts need to be explicitly assigned when territories are configured and activated in the system. This is one exception which does not have immediate/automatic assignment. The recommendation is to run a batch assignment to assign these sales accounts created at the beginning of the implementation using the view criteria SalesAccountsUpdatedSinceVC.

Partner territories get assigned to sales accounts in the following scenarios.

• When a partner-generated lead is approved, all partner territories associated to the partner-generated lead are automatically assigned to the sales account.
• Users with the privilege Manage Sales Party Partner Territory can assign partner territories from the sales account team UI.

Note
Territory Management must be implemented to utilize this feature.

How can I make merge requests?

A merge request is made when duplicate records that point to the same customer are found, and you want to consolidate those records into one. When a merge request is approved, there is one survivor record. All other duplicate records are considered victims, and they are marked with the status of Merged. In customer center, you can mark two or more customer records for merge request from customer list in the Customer home page or in the customer search result. Merge requests will be processed by the customer data hub. The customer data hub must be implemented and the profile option Merge Request Enabled set to YES for this feature to be available.

What happens if I include a free-form response for a question?

A score of 0 is assigned for free-form responses.

A free-form response option will have no effect on the overall assessment score. The free-form response offers the opportunity to enter a textual response to a question that does not conform to any of the pre-populated responses provided by the assessment template.

What's a Question Group?

A question group is a logical grouping of questions within an assessment template, and it is used strictly as a category header for those questions. Through
careful naming of a question group, you can achieve the benefit of providing the user of the template with an approximate idea of the type of questions to expect in each group.

**Why am I being asked to enter question weights again?**

This step lists all of the assessment template questions in one place, and provides you with the opportunity to edit weights as necessary to ensure that the sum of all weights totals 100.
Define Territory Management Configuration

Territory Components: How They Work Together

Territories are used to define the jurisdiction of responsibility of a salesperson over a set of sales accounts. Sales managers use territory proposals to change territory definitions. Managers can create more than one territory proposal and use metrics and graphs to compare and analyze their proposed territories for fairness, effectiveness, and alignment with current sales goals. Managers then activate the best territory proposals.

This figure shows the use of territory proposals to add, change, and delete territories. After analysis, managers activate final territory proposals.
Territories

A territory, whether active or part of a territory proposal, includes several elements. One or more dimensions, such as geography, define the boundaries of a territory according to selected dimension members, such as Europe or Asia. Every territory is assigned an owner and can have additional territory team members.

This figure shows two territories defined using the same two dimensions but different dimension members. Each territory has an owner and a sales team.
Define Territory Management Dimensions

Territory Dimensions: Explained

Dimensions are attributes that define jurisdiction boundaries of territories. For example, the geography dimension can be used to define territories by country or postal code. Territory dimensions are used to assign sales accounts, leads, and opportunities to the correct territories.

Each territory dimension is matched to an attribute of the object being assigned. Product and Sales Channel are matched to lead and opportunity attributes directly. The rest of the dimensions are matched to sales account attributes either when assigning sales accounts to territories, or when assigning leads and opportunities to territories, in which case the sales account for the transaction is used. All dimension values combine to define the territory boundaries. For example, if Geography = United States and Product = Green Servers, then the territory boundaries are United States AND Green Servers.

The dimensions available for defining territories are:

- Account
- Account type: Named or Not Named
- Customer size: from the Organization Size lookup
- Geography
- Industry
- Organization type
- Product: A hierarchy from the Sales Catalog
- Sales channel
- Auxiliary 1, 2, 3
Your administrator enables the dimensions your organization uses for defining territories. The administrator also selects the dimension members that appear in the selection list when defining territory dimensions. All invisible dimension members appear in an other category in the selection list. An Unspecified dimension member captures objects with insufficient matching values.

**Account**

Sales accounts and their hierarchies are maintained in the Oracle Fusion Customer Center application.

**Geography**

The geography hierarchy is maintained using the Define Territory Management Zones in Oracle Fusion Trading Community Model.

**Industry**

The industry hierarchy is from the customer classification module.

The profile option Industry Classification Category must be set to the classification category that you want to use before you can enable the dimension. The available selections include only classification categories belonging to the Industrial Categories grouping.

**Organization Type**

The organization type hierarchy is from the customer classification module.

**Sales Channel**

The list of sales channels (Direct and Partner) and their types are from Oracle Partner Management.

**Auxiliary 1, 2, 3**

There are three dimensions from the customer classification module that you can use to create your own dimension names.

The profile option Classification Category for Auxiliary Dimension 1 (2, 3) must be set to the classification category before you can enable the dimension. The available selections include only classification categories belonging to the Customer Categories grouping. The classification category you choose in the profile option becomes the display name for the dimension.

---

**Manage Territory Management Synchronization**

**Dimension Parameters: Explained**

Use parameters to refine the definition of each dimension so it best fits business requirements.

Three dimensions use parameters:

- Industry
- Time
• Product

**Industry**

Use the parameter to set the number of levels in the industry hierarchy that display in the dimension member selection screen.

**Time**

Set the start of the time period used in calculating metrics using the Calendar Start parameter. Set the number of years prior to today’s date when the calendar starts for metrics.

**Product**

Use the parameter to set the number of levels in the product hierarchy that display in the dimension member selection screen.

---

**Populating Dimension Members: Explained**

Dimension members are populated and synchronized in Oracle Business Intelligence Suite Enterprise Edition Plus either via the data warehouse, or without the use of data warehouse.

**Non-Data Warehouse Option**

When you do not use data warehouse, dimension and transaction data are populated directly from Fusion Applications into territory dimension members and the Oracle Essbase cube. The structure of the cube reflects the enabled dimensions in Territory Management and their members. The cube provides metrics information for defined territories.

This figure shows the population of the dimension members data and the Oracle Essbase cube using the Repository Project Design (RPD) view from Business Intelligence.
Data Warehouse Option

In the Data Warehouse option, the data warehouse Extract-Transform-Load (ETL) logic extracts transaction and dimension data from Oracle Fusion Applications into the data warehouse first. The data warehouse stores dimensions and metrics data. Metrics data is based on transaction information. Territory Management contains only a subset of the dimensions stored in data warehouse.

This figure shows the use of ETL to load data from Fusion Applications to the data warehouse, and RPD to provide data to the cube and to Territory Management. ETL also loads the settings for visible and invisible dimension members from Territory Management to the data warehouse.

Use ETL to load data to the data warehouse when:

- There are changes in the dimension members sources (Fusion Application), so that the new structure of the dimensions is reflected in the data warehouse and consequently in Territory Management.
- After changing visibility settings for dimension members.

Cube

Territory Management generates the cube based on the enabled dimensions and their members. Territory Management loads the cube with dimension members
and metrics directly from the data warehouse, in the case of the data warehouse option, or from Fusion Applications, in the case of the non-data warehouse option. The cube provides metrics information for defined territories.

**Sequence**

There are several aspects to enabling and populating dimension members, including setting profile options for certain dimensions.

Following is the sequence for populating dimensions for use in defining territories:

1. Change profile options for industry classification category or the classification category for auxiliary dimensions 1, 2, or 3.
2. If you are using data warehouse, then run ETL to populate the data warehouse with the dimension members and transactional data.
3. In Territory Management, identify the dimension members to be made visible in selection lists for defining territories. Refresh the members before making visibility selections.
4. If you are using data warehouse, then run ETL to populate the data warehouse with visibility settings.
5. Use the stage environment in Territory Management to enable dimensions. Execute the Stage action which synchronizes the Territory Management stage environment with the data source. You can start defining territories using newly enabled and populated dimensions after you promote the stage environment to production.

**Using Territory Dimensions: Examples**

The sales administrator enables only the dimensions the organization requires for defining territories. The following examples illustrate the use of different dimensions to assign sales accounts, leads, and opportunities to the correct salespeople using defined territories.

**Geography**

For most of your sales activities, you want to assign salespeople by city and postal code.

**Account**

You have a few key accounts that should belong to top salespeople. Use the account dimension to create territories for individual sales accounts.

**Account Type**

You want to assign major sales accounts to Named accounts territories A named account territory can have child territories identified by other criteria, such as geography. You also have territories with the account type of Not Named that include no major named sales accounts in the hierarchy.

**Customer Size**

One product line is suitable only for organizations above a certain size. Use the customer size dimension to target only the larger customers for the product line.
Industry

You sell one type of service to telecommunications companies, another service to utilities, and a third service for insurance companies. You can create territories for each using the industry dimension.

Product

You sell a product line that requires salespeople to have a high degree of technical knowledge. Create separate territories for this product line.

Sales Channel

You delegate sales accounts that are small to partner sales organizations by geography.
How Lead Components Fit Together

A lead follows a path which ends either with converting the lead to an opportunity, or retiring the lead when no possibility exists of converting the lead to a sales opportunity. The lead lifecycle includes an automated process to first capture the leads, then prioritize the leads for sales engagement through a scoring and ranking process. Leads are then distributed to appropriate sales resources for further lead qualification, follow-up and conversion.

**Leads Lifecycle**

Leads are monitored, reassigned as appropriate and the lead quality is continuously reviewed and adjusted as the lead progresses through its lifecycle. Marketing and Sales departments both share the ownership of leads, where the focus on the leads shifts from Marketing to Sales and back to Marketing based on the lead status. The lead lifecycle is captured in the following topics:

- Lead Generation
- Lead Qualification
- Lead Distribution
- Lead Assessment
- Lead Conversion

**Lead Generation**

Leads are generated and captured from many different sources such as:

- Campaign responses
- Campaign stages handled by telemarketing
- Third-party lead sources
- Sales prediction application through the creation of new leads

Flexible lead import, customer and contact creation, and de-duplication ensure marketing lead generation efforts are optimized. For example, the lead import process checks whether leads represent new or existing customers. For new customers, data needs to be created for the lead. If the lead is an existing customer, part of the lead import process checks to ensure customer and lead information is not duplicated.

**Lead Qualification**

Marketing departments help with the lead qualification process to ensure that only qualified leads are handed over to sales. Leads are typically ranked as Hot, Warm, or Cool. Leads are further qualified by the use of company specific standard questions to score a lead. Lead scores are numeric values typically ranging from 1 to 100, where a high score represents high quality.

It is not good practice to let stale leads build up. Standardized criteria for lead qualification ensure that quality leads reach the salesperson and help maximize the conversion rate from leads to opportunities. For example, your organization has criteria and processes for ensuring that leads are either developed or retired.
within 30 days. When the lead age is greater than 30 days and the rank is A or B, Marketing reassigns the leads for follow-up by an internal telemarketing group. If the leads cannot be qualified or further developed to revenue opportunities, the rejected leads can be reassigned or can be retired manually.

**Lead Distribution**

As the qualification of leads progresses into real potential prospects, assignment manager uses expression-based rules to associate one or more internal salespersons with each lead. If the lead is associated with an existing Sales Account, then assignment manager uses territory definitions to associate (typically one) internal territory with each lead. The salesperson newly assigned to the lead may be related to the lead record directly through the lead team or indirectly through a territory associated with the lead. They can view and update those leads to which they are assigned in the lead work area and can claim ownership of the lead by using the Accept Lead action. Other assigned resources can view and update the lead, but cannot make themselves the owner. As the lead is qualified further, for example, if a sales prospect changes to a sales account by adding an address, assignment manager is automatically invoked during the next automated assignment cycle. Depending on the assignment logic, the lead may be reassigned to a different territory or sales resource. If the assigned salesperson takes no action on a lead for several days, then the lead can be manually reassigned to another salesperson.

**Lead Assessment**

The salesperson must evaluate the quality of information they have received for the lead. They determine if the details are sufficient to reach out to the customer and assess whether a lead is worth pursuing with the help of preconfigured assessment templates. Assessment templates can further qualify the lead by:

- Reviewing the content shared with the customer during a campaign
- Framing the lead in the context of the campaign
- Ensuring the salesperson understands the information that has already been sent to the customer

Lead Assessment enables leads to be further assessed through predefined questions that help determine the likelihood of the lead being accepted by Sales. In this scenario, a salesperson named Mike begins asking the customer a series of questions created by Marketing and Sales to assess the quality of the lead. As each question is asked, Mike records the answer and the lead assessment tool automatically factors the answer into the assessment score of the lead. At the end of the call, Mike notes that the assessment lead score is high. He requests the lead be assigned to the direct sales team for that customer. If the lead score was low, then Mike could retire the lead, or if the lead needed further qualifying, he could leave it in his list of leads for follow-up at a later date. If the lead is good, but the potential revenue opportunity is less than a predetermined monetary amount, for example, twenty-five thousand dollars, then Mike can convert the lead to an opportunity that he works himself.

**Lead Conversion**

A lead’s life cycle ends either when a lead is converted to a sales opportunity, or when the lead is retired. Conversion to an opportunity stage allows the salesperson to pursue the account in the sales cycle. After establishing that the lead has potential, the salesperson converts the lead to an opportunity. Contact is established and meetings and presentations are scheduled to move the opportunity along the sales pipeline. To track the progress, contact notes are captured as interactions and associated with the contact and opportunity.
As the lead progresses through its life cycle, decisions to retire the lead are based on the following.

- You cannot verify customer and lead details
- The customer is not interested in pursuing the lead any further

Configure Assignment Manager for Lead Processing

How Assignment Object Components Work Together

The work object, candidate object, and attributes are components that fit together to create assignment objects that are used in rule-based and territory-based assignment. Work objects are business objects that require assignment such as leads and opportunities. Candidate objects are business objects such as resources and territories that are assigned to work objects.

When you create candidate objects, you can select attributes for them that are later used in rules or mappings. These candidate objects also become candidates that are available for association when you create work objects. When you create work objects, you can select attributes for them also, as well as associating one or more candidates.
Work Object

A work object is a business object that requires assignment such as a lead or an opportunity. Creating a work object involves entering its application information, selecting its attributes to use during assignment, and associating one or more candidates.

Candidate Object

A candidate object is a business object such as a resource or a territory that is associated with one or more work objects for eventual assignment. Creating a candidate object involves entering its application information and selecting its attributes to use in rules or mappings. A special type of candidate object is a classification object. This type of candidate object does not represent a business
object that gets assigned to a work object. It is used only with classification rules and is used primarily to rank or qualify leads.

**Note**

As candidate objects are created, they become available as candidates that can be associated with one or more work objects as part of the work object creation process.

**Attributes**

Attributes are elements in the view object defined for an assignment object. For each assignment object, you can select one or more attributes that you want to use when configuring assignment rules or mappings. For example, for a work object like sales account, you might choose the attributes of Named Account Flag, Customer Size, and Organization Type. When you configure assignment rules for the sales account work object, your chosen attributes are available for your rule conditions. In other words, you could configure a rule for sales account using the Named Account Flag attribute, and set a condition where the assignment engine looks for sales accounts that have their Named Account Flag equal to Yes.

When selecting attributes for a candidate object, you will not only select the attributes you want to use when configuring assignment rules and mappings that involve that candidate object, but you also want to select the attributes for that candidate object that you want to appear in the screen that displays recommended candidates after assignment manager is run. For example, if a candidate object is resource (sales representative), and you want to show sales representatives’ first names, last names, and phone numbers when they are recommended during assignment processing, you need to select the attributes for the resource candidate object that correspond to first name, last name, and phone number, and you need to specify the order in which these attributes will appear in the recommended candidates screen.

**Note**

This feature is not used by any CRM applications at this time.

**Adding an Assignment Object**

The Manage Assignment Objects pages enable you to define and edit the Work and Candidate objects as well as define any territory-based mappings. The figure above shows the relationship between the work and candidate objects and the mapping of the matching candidates to work objects.

When you add or edit a work or candidate object there are several key pieces of information that are required in the definition:

- **Name**: a unique name for the object with an optional description.
- **Code**: a unique code used in processing the object.
- **Work/Candidate Object check boxes**: indicates if the object is a work object, candidate object or both.
- **Application Module**: an Oracle Application Development Framework (ADF) business component that encapsulates the business service.
methods and UI-aware data model for a logical unit of work related to an end-user task. Enter the fully qualified definition name of the consumer application, Application Module. Valid for top level Work and Candidate objects. Child objects automatically inherit this value from its parent.

- Application Module configuration: Valid for Top Level Work and Candidate objects except Classification Candidate objects. Child objects will automatically inherit this value from its parent.

- View Object Instance: used to define the data model of a view object component when designing an application module, for example, Opportunity. Valid for all levels of Work and Candidate objects except Classification Candidate objects.

- View Criteria may be defined to filter the information for the rows of a view object collection. Valid for top level Work and Candidate objects except Classification Candidate objects.

- Primary Key Attribute 1: First or only attribute that makes up the object primary key. Valid for top level Work and Candidate objects except Classification Candidate objects.

- Refresh Interval: the number of minutes between refreshes of candidate object data. The default setting is 0 minutes. Valid for top level Candidate objects except Classification Candidate objects.

- Initial Caches: The initial size of the cache when processing an object. This value will be used the first time the engine processes objects or following a server bounce. The default value is 2, and the maximum value is 20. Only valid for top level Candidate objects except Classification Candidate objects.

Note

All Work Objects that are used for scoring, Lead for example, use the Product Level (MOW_SCORING_INITIAL_CACHES) Initial caches for scoring rules profile option value.

- Maximum Caches: The maximum size of the pool/cache when processing the object. The default value is 5, and the maximum value is 25. Only valid for top level Candidate objects.

Note

All Work Objects that are used for scoring, Lead for example, use the Product Level (MOW_SCORING_MAX_CACHES) Maximum caches for scoring rules profile option value.

- Score Attribute: The attribute on the object that stores the total calculated score after an assignment request has been processed. Valid for top level Work objects only.

- Assignment Date Attribute: The attribute on the object that stores the assignment date after an assignment request has been processed. Valid for top level Work objects.
• Exclude Assignment Attribute: The attribute on the object that stores the setting for excluding a work object from assignment. Valid for top level Work objects.

Adding Assignment Object Attributes

Assignment Manager allows users to specify a set of attributes from the assignment object VO to be used during the assignment evaluation. The assignment engine will load these Assignment Object Attributes for each assignment object VO row, in addition to any primary key or assignment attributes. This is designed to improve performance by not loading those attributes not used for the assignment evaluation.

Assignment Object Attributes should be defined for each work object and any child objects as well as each candidate object to be used by the Assignment Engine.

• View Object Attribute: Name of each attribute in the view object defined for the assignment object. Assignment Rules or Mappings can be configured using these attributes. For Candidates Objects, the attributes that appear in the interactive assignment UI should also be selected.

• Candidate Information Sequence: The sequence that this attribute is displayed in the Interactive Assignment UI.

Defining Work Object and Candidate Object Associations

The administrator needs to define the association between the work object and candidate object. For example, the Lead work object may have an association with both the Territory candidate object and the Resource candidate object. This implies that Assignment Manager can be used to assign Territories and Resources to a lead.

Configuring Assignment Manager: Critical Choices

Assignment is the process for selecting a candidate as an object and executing the association to a work object. Assignment consists of two phases. The first phase is the matching phase, where matching rules or mappings are evaluated to find the right assignees from a list of possible candidates. The second phase is the disposition phase, where the disposition, or assignment, of matching candidates is handled. Assignment Manager is the tool used to establish the business objects that require assignment, to set up the resources that can be assigned, and to create the rules and mappings that dictate the selection and assignment of those resources. Candidates are potential assignees for a work object. A work object is a representation of an application business object inside Assignment Manager. A work object captures the attributes of a business object and associated child objects to be used for matching purpose. To best plan the configuration of Assignment Manager, you should consider the following points:

• Business objects
• Resources
• Assignment disposition
• Attributes
• Mappings and rules

Business Objects

A business object is a data entity or a collection of data treated as a unit, such as a sales account, an opportunity, or a lead. Any business object that requires the assignment of a resource to act upon it is considered a work object by Assignment Manager. The work object is a representation of the business object, and mappings and rules are developed to ensure timely and accurate assignment of candidates (for example, territories or resources) to those work objects. When configuring Assignment Manager, carefully consider which of your business objects require assignment, and create work objects only for those that do.

Assignment Disposition

After you determine the business objects (work objects) that require assignment and the candidate objects that you will assign to them, you must decide how the matching candidate assignment disposition will be carried out. Consider these questions:

• Do you want to assign a single resource or multiple resources?

• Do you want to automatically assign matching candidates or run custom logic against matching candidates?

• Do you want to record the matching candidate score on the work object?

• Do you want to retain manually assigned candidates when assignments are processed?

• Do you want to replace disqualified candidates when assignments are processed?

Attributes

To ensure that candidates are properly assigned to work objects, you will create mappings and rules. These mappings and rules employ attributes to determine the best assignments. As you set up work objects and candidate objects in Assignment Manager, you will also select the attributes of those objects that you want to use in your mappings and rules. For example, you might want to assign a resource such as a sales representative to a business object like opportunity based on the product skill of the sales representative. In this case, when you create the opportunity work object and the sales representative candidate object, you will select the attributes of opportunity and sales representative that correspond with product skill. Selecting these attributes makes them available for mappings and for conditions on your rules, so ensure that you select the attributes that reflect the criteria that you want to use for matching business objects to work objects.

Mappings and Rules

Assignment mappings drives territory-based assignment. These mappings identify the dimensions, attributes, and territory filtering used in territory-based assignment processing. A default set of mappings are seeded. This seeding assumes that opportunities, leads, and sales accounts use the same territory
hierarchy. Rules are defined for the execution of rule-based assignment. Rules are designed to return candidates based on whether these candidates match a set of criteria, are within a defined scoring range, or are of a specific classification.

You create the mappings and rules using the work objects, candidate objects, and attributes that you already established. When designing your mappings and rules, carefully consider how you want to match candidates to work objects. For example, would you want resources assigned based on their geographic location, or their product knowledge, or their skill level, or a combination of any of these attributes? Do you want to match candidates only, or would you like to match candidates and score them? In a multiple-candidate scenario, do you want to assign all matching candidates or only those who achieve higher than a specific score? Consider these questions before creating mappings and rules.

Creating Assignment Mappings: Examples

For territory-based assignment, you create work-object-to-candidate-object mappings during assignment object creation. These mappings are used to make candidate assignments. You can create multiple types of mappings for assignments. The following scenarios illustrate these different mappings:

- Creating an attribute mapping
- Creating a dimension mapping
- Creating a literal mapping

Creating an Attribute Mapping

You want to assign territories to a sales lead when the territory program ID is the same as the sales lead program ID. Create a mapping where the work object is sales lead and the candidate object is sales lead territory. Select the territory when the attribute territory program ID is equal to the sales lead attribute program ID.

Creating a Dimension Mapping

You want to assign territories to opportunity revenue lines based on the product associated with the revenue line. Create a mapping where the work object is opportunity revenue line, and the candidate object is territory. Select the product dimension as the mapping type. The candidate object low and high attributes correspond to the names of the low sequence and high sequence attributes for product on the territory. The work object low and high attributes correspond to the names of the low sequence and high sequence attributes for product on the revenue line. For example, the low sequence attribute for product on the revenue line might be called ProdSeqLow.

- Mapping using alternative attributes: Using the same scenario of assigning territories to opportunity revenue lines based on the product associated with the revenue line, you might encounter a situation where a revenue line does not have a product assigned to it, but it does have a product group assigned to it. Create the same mapping that you created for the dimension mapping scenario, and add the names of the low sequence and high sequence attributes for product group for the work object alternate low and high attributes. For example, the alternate low
sequence attribute for product group on the revenue line might be called \texttt{ProdGrpSeqLow}.

- Mapping using default values: Using the same scenario of assigning territories to opportunity revenue lines based on the product associated with the revenue line, you might encounter a situation where the low sequence and high sequence attributes for product on a revenue line do not contain values when assignments are processed. Create the same mapping that you created for the dimension mapping scenario, and add low and high default values for the product attribute for revenue lines.

**Creating a Literal Mapping**

Literal mappings are a way of filtering the matched territories based on specific values of a territory attribute. You want to find only territories that are finalized (for example, territory status equals \texttt{FINALIZED}).

**Lead Assignment: How It Is Processed**

Once lead data is cleansed, created, enriched, and scored, the leads need to be assigned. Leads can be assigned based on several criteria. For example, you can configure Assignment Manager to assign leads based on the lead source, geography, named accounts, (such as the top 20), industry, product, partner (for working with a partner organization), and primary sales channel associated with the lead or associated marketing campaign that generated the lead.

**Components That Affect Lead Assignment**

The following lists the components that influence the assignment of leads:

- Lead work objects
- Lead candidate objects
- Attributes
- Mapping and rule conditions

**How Leads Are Assigned**

Lead Assignment process evaluates both rules as well as territories for lead assignment. Since leads on sales prospects cannot be distributed based on territory definition, they are assigned based on simple rules evaluation. Territory-based evaluation can be supplemented by adding filtering rules to further refine the lead assignment. The territory-based evaluation component uses the lead work object and territory candidate object data, and the mappings between the territory dimensions and lead dimensional attributes to execute the assignment processing.

Set up the following for assigning leads:

- Set up the necessary lead work objects, and associated candidate objects to be assigned by Assignment Manager.

\textbf{Note}
A default set of lead work objects and associated candidate objects are seeded.

- Set up rules and rules set specific to your business requirements. For example, set up an assignment rule to assign leads with deal size less than a certain amount to partners.

- Set up object mappings for territory-based assignment. For example, assign a lead to those territories where Territory dimensional attributes are mapped to corresponding Lead attributes. Territory-based assignment relies on an association between attributes on the lead work object and attributes on the lead territory candidate object to match the candidates to the work object.

- Set up rules to filter territories that match the lead based on additional information on the lead. For example, set up rules to exclude prime sales territories that match leads which are unqualified.

For territory-based assignment, the lead work object and lead candidate object must have attributes that share the same domain of values. The mapped attributes are used for matching appropriate candidates for a work object. For example, a sales lead (work object) has a geographic location attribute. The lead candidate object (territory) has also a geographic location attribute. These two attributes are mapped to each other.

**Disposition Logic: How It Is Processed**

Assignment Manager, a part of Oracle Fusion Management of Work, uses a two-step process to assign candidates. The first step is the selection of Candidates with the Assignment Manager application. Candidates to be assigned to the work object are determined by territory-based mapping, rule-based assignment, or territory-based mapping with rule-based filtering.

Disposition is the second step. The selected candidates are written to the work object table or the work object child table based on the selected disposition logic. This has several possible outcomes, but it is based on two principal logic features. The first is the delivered Assign Candidate logic. The second is Run Custom Logic, a way for you to define the assignment disposition of the candidate object. For example, you may want to e-mail a specific list of candidates, or write the selected candidates to a specific table. Run Custom Logic can be used in conjunction with the Assign Candidate logic.

**Settings That Affect Disposition**

By designating Assign Candidate, Assignment Manager can work with one or multiple candidates that have been returned in the selection process.

- Single Candidate Returned: Assignment Manager writes the candidate back to the work object table based on the primary key attribute fields that you must enter for that work object table.

- Multiple Candidates Returned: A work object child table is designated to record these candidates. Then you indicate at least one and up to three attributes that serve as the primary key fields. There are two choices for
how to write the resulting candidates, Replace Candidates and Keep Manual Candidates.

- Replace Candidates: When the Assign Candidates logic is run, the results are compared to any results already in the work object child table. If Replace Candidates is checked, any existing candidates are removed and replaced with the new results. No manually entered candidates will be affected.

- Keep Manual Candidates: Some applications using Assignment Manager allow manual entry of candidates. When a candidate is entered manually, there is an attribute that indicates manual entry that is set to Y or Yes. When Keep Manual Candidates is checked and the manual attribute indicated, Assign Candidates logic will disregard these candidates when writing to the work object child table.

Selecting Run Custom Logic allows you to develop your own code to perform specific actions on or with the candidates that are returned from the selection process. When this option is chosen, after the selection process, Assignment Manager will use a callback function in the application module which will call your custom logic code and pass on the information received from the selection process.

**How Disposition Logic Is Processed**

Assignment manager selects the candidates and then passes in the results from the selection phase to either the assign candidate or custom logic or both.

If you check both Run Customer Logic and Assign Candidates on the user interface, the assignment engine will invoke the Assign Candidates logic first and then execute the Custom logic.

---

**Note**

If you are using credit allocation templates in Oracle Fusion Opportunity Management, you may wish to leave both Run Custom Logic and Assign Candidate unchecked. Opportunity Management uses Assignment Manager to find matching credit allocation templates. In this process, Assignment Manager passes the template IDs back to Opportunity Management which does the assignment. No custom logic is involved and Assignment Manager does not perform the assignment.

---

**Using Territory-Based Assignment with Rule-Based Filtering: Example**

In this example sales leads with sales accounts can be assigned one or more territories and supplemental lead team resources. Prospect leads can be assigned one or more resources. Assignment Manager is used to determine matching territories as well as the matching resources. In Oracle Fusion Partner Relationship Management (PRM) implementations, all territories (Prime, Overlay, Partner, etc.) matching a given lead may be identified and rule filtering then used to affect the type of territories (Partner versus Prime) that are assigned.
based on the value of specific attributes, for example sales channel or deal size, on the lead.

Scenario

Acme, Inc., wants to assign new leads to the correct territory as well as then assign them to the correct Sales Lead. If there is no Sales Channel assigned, determine if the deal should go to a partner or remain internal and assigned to a Channel Manager.

Transaction Details

The primary Marketing business objects processed by the Assignment engine are Leads. The assignment of territories is the primary means of assigning the appropriate sales people to the lead. Rule filtering may also be used to filter the territories when the sales channel is not identified. Prospect leads are processed by the assignment engine in order to identify additional resources based on the information on the leads (such as deal size)

Oracle Fusion Lead Management calls the Find Match and Assign API, with the Work Object designated as Lead and territory as the Candidate Object with the assignment type of territory-based assignment. This determines the list of territories. The API then calls the Rule Set Group that contains the rules for the territory-based assignment with rule-based filtering.

While the territory-based assignment delivers a list of territories, the rules can fine tune the assignment process:

1. Rule for SALES CHANNEL = NULL
   a. SalesLead.Sales Channel =! NULL
2. RULE for SALES CHANNEL = NULL, Assign Partner
   a. Sales Lead.Sales Channel = NULL
   b. Sales Deal.Deal SizeAttribute < 1,000,000
   c. Territory.Territory Type = Partner
3. RULE for SALES CHANNEL = NULL, Assign Channel Manager
   a. Sales Lead.Sales Channel = NULL
   b. Sales Deal.Deal SizeAttribute < 1,000,000
   c. Territory.Territory Type = Sales Channel Manager
4. RULE for SALES CHANNEL = NULL, Assign Channel Manager
   a. Sales Lead.Sales Channel = NULL
   b. Sales Deal.Deal SizeAttribute > 1,000,000
   c. Territory.Territory Type = Prime

Analysis

When the lead comes in, it needs to be assigned to a territory for follow up. Based on the above rules, you can determine is this is a smaller deal that can be handled by your partners, or it is a larger deal that needs to be followed up by an in house Channel Manager.
The assignment manager selects the first list of territories for the lead. The rules then determine who gets the deal:

1. This first rule determines if a Sales Channel value exists. If not, it is assigned to the Partner.
2. The second rule says if there is no Sales Channel assigned and the deal is under one million dollars, assign it to a Partner.
3. The third rule assigns an internal channel manager if there is no Sales Channel and the deal is under one million dollars.
4. and the final rule is for a PRM implementation. If there is no Sales Channel defined, and the deal is greater than one million dollars, keep the deal internal as the Prime scenario.

**Territory Lead Assignment: Examples**

Leads are assigned to the appropriate territories based on matching lead attributes to territory dimensions.

A territory is the jurisdiction of responsibility of a sales resource over a set of sales accounts. Use territory-based assignment to assign sales territories to leads.

**Assign territories to lead territory team**

Assignment Manager can be set up to automatically assign sales territories to the lead using territories defined in Territory Manager. For example, a salesperson navigates to the leads list and opens the newly created lead. Using the Reassign action, the salesperson invokes Assignment Manager and selects an option to run automatic assignment immediately to reassign the lead to the appropriate territories.

**Territory-based assignment mapping**

Territory-based assignment mappings are predetermined and are available as part of the Lead Management functionality. Predefined mapping are leveraged for matching the correct territories with each lead. For example, you can choose to map the location attribute on the lead to the geography attribute on the territory. Any territories where the geography value matches the location of the lead is matched and assigned the lead.

**Analysis**

Ensure you have defined your territory boundary based on dimensions. There is only one set of mappings for a work object and candidate object combination. The mappings for various assignment scenarios (such as assignment of a territory to a lead) must be setup through a mapping that assigns appropriate sales territories to a lead territory team. If the lead needs rule-based assignment, then rule sets are used. If the lead needs territory assignment only, then the rules may not need to be used.

You can enable only the dimensions that your organization requires for defining territories. The following examples illustrate different dimensions used to assign sales leads to the correct sales territories.
<table>
<thead>
<tr>
<th>Dimension</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>For most of your sales activities, you want to assign salespeople by city and postal code.</td>
</tr>
<tr>
<td>Account</td>
<td>You have a few key accounts that should belong to top salespeople. Use the account dimension to create territories for individual sales accounts.</td>
</tr>
<tr>
<td>Customer Size</td>
<td>One product line is suitable only for organizations above a certain size, and you have a few skilled salespeople for that product line. Use the customer size dimension to assign skilled salespeople to the larger customers for the product line.</td>
</tr>
<tr>
<td>Industry</td>
<td>You sell one type of service to telecommunications companies, another service to utilities, and a third service for insurance companies. You can create territories for each using the industry dimension.</td>
</tr>
<tr>
<td>Product</td>
<td>You sell a product line that requires salespeople to have a high degree of technical knowledge. Create separate territories for this product line.</td>
</tr>
<tr>
<td>Sales Channel</td>
<td>Your sales department prefers to engage partners as indirect sales channel, and telesales functions in addition to the direct sales force. You can create territories for these different sales channels such as telesales, direct, and indirect sales channels.</td>
</tr>
</tbody>
</table>

**Lead Assignment Objects: Explained**

Fusion Lead Management provides preconfigured work objects, candidate objects, and attributes to automate the process of assigning leads to salespersons, partners, sales resources, and sales territories through assignment manager.

Lead assignment selects a candidate for a work object and processes the association. You create rules with conditions that need to be met in order to make the appropriate candidate assignments. This is called rule-based assignment. Alternatively, you create attribute to attribute mappings between a work object and a candidate object in order to make the appropriate territory candidate assignment and this process is called territory-based assignment.

This topic outlines the following:

- Lead Work Objects
- Candidate Objects Available for Lead Assignment
- Lead Assignment Attributes
- Score Attribute

**Lead Work Objects**

The lead work object is a representation of the lead business object. Creating a lead work object involves entering application information, selecting attributes to use during assignment, and associating one or more candidates. The Lead Management application has preconfigured the following lead work objects to ensure timely and accurate assignment of territories or resources to leads:
When implementing the Lead Management application, consider which leads require resource assignment. For example, you might decide that you do not want to assign resources to leads that are less than 100 dollars in value, but want to assign resources to leads that have a deal size of 100 dollars or more.

### Candidate Objects Available for Lead Assignment

A candidate object is a business object such as a resource or a territory that is associated with one or more work objects for eventual assignment. As candidate objects are created, they become available as candidates that can be associated with one or more work objects as part of the work object creation process. Lead candidate objects are resources, such as a salesperson, or might be a virtual entity such as a territory, that are assigned to a lead work object. The Lead Management application has preconfigured the following lead candidate objects to ensure timely and accurate assignment of territories or resources to leads:

<table>
<thead>
<tr>
<th>Lead Candidate Object Name</th>
<th>Represents ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales Lead Territory</td>
<td>A territory evaluated for assignment to the sales lead team.</td>
</tr>
<tr>
<td>Sales Lead Resource</td>
<td>An individual resource evaluated for assignment to the sales lead team.</td>
</tr>
<tr>
<td>Sales Lead Rank</td>
<td>The rank value to assign to the sales lead based on rules.</td>
</tr>
<tr>
<td>Sales Lead Qualification Status</td>
<td>The qualification status value to assign to the sales lead based on rules.</td>
</tr>
</tbody>
</table>

Creating a candidate object involves entering application information and selecting attributes to use in rules or mappings. Consider the following questions when you enter candidate details and disposition information for your lead work object:

- Assign a single resource or multiple resources?
- Automatically assign matching candidates or run custom logic against matching candidates?
- Record matching candidate score on the work object?
- Replace disqualified candidates when assignments are processed?

### Note

A special type of candidate object is a classification object. This type of candidate object does not represent a business object that gets assigned to a work object.
It is used only with classification rules and is used primarily to rank or qualify leads.

You can choose to select multiple candidates and exclude other candidates when setting up assignment manager for leads. If you want to assign more than one candidate, select the combined work object and candidate object view instance that captures information for the candidate.

For mapping-based lead assignments, you can select from the following generation types:

<table>
<thead>
<tr>
<th>Generation Type</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension</td>
<td>Create dimension mappings between a work object and a candidate object in order to make the appropriate candidate assignment. In this scenario, a mapping is created between an attribute on the lead work object and the territory candidate object. The mapping is set up to map the Primary Product attribute on the lead to the Product attribute on the territory. Any territories where the Product Dimension value matches the Primary Product on the lead will be matched and lead is assigned to that territory.</td>
</tr>
<tr>
<td>Literal</td>
<td>Create literal mappings on a candidate object in order to make the appropriate candidate assignment. In this scenario, a literal mapping is created on the territory candidate object to exclude territories that are not Active. As a result, any territories which are not Active will not be matched.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Create attribute to attribute mappings between a work object and a candidate object in order to make the appropriate candidate assignment. In this scenario, a mapping is created between the lead work object and the territory candidate. The mapping is set up to map the Partner attribute on the lead to the Partner attribute on the territory. Any territories where the Partner value matches with the Partner on the lead will be matched and lead is assigned to that territory.</td>
</tr>
</tbody>
</table>

**Lead Assignment Attributes**

Attributes are elements in the view object defined for an assignment object. To ensure that resources are properly assigned to business objects, you create mappings and rules. These mappings and rules employ attributes to determine the best assignments. As you set up your lead work objects and lead candidate objects in Assignment Manager, choose from the preconfigured attributes for those lead work and candidate objects that you want to use in your mappings and rules.

For example, you might want to assign a candidate such as a salesperson to a lead work object based on the geographical location of the salesperson. In this case, you select the attributes of the lead work object and the salesperson candidate object that correspond with geographical location. Selecting these attributes makes them available for mappings and for conditions on your rules,
so ensure that you select the attributes that reflect the criteria you want to use for matching resources to business objects.

The following table shows the preconfigured lead candidate objects and associated lead attributes excluding ID attributes.

<table>
<thead>
<tr>
<th>Lead Candidate Objects</th>
<th>Associated Lead Attributes (ID’s excluded)</th>
<th>Updated Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Territory</td>
<td>• ChannelType</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ClassCode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DecisionMakerIdentified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• OrganisationSize</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• PartnerChannel</td>
<td></td>
</tr>
<tr>
<td>Resource</td>
<td>• PartnerType</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rank</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Score</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SourceCode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Status</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• StatusCd</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Timeframe</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ToReassignFlg</td>
<td></td>
</tr>
<tr>
<td>Lead Qualification</td>
<td>• PartnerType</td>
<td>StatusCd</td>
</tr>
<tr>
<td></td>
<td>• Rank</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Score</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SourceCode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Status</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• StatusCd</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Timeframe</td>
<td></td>
</tr>
</tbody>
</table>

**Score Attribute**

Score attribute is used to store the calculated score after a scoring request (based on Assignment Manager) has been processed.

Lead scoring associates a value to each lead based on weighted criteria and scoring attributes appropriate to your organization. It might be as simple as scoring based upon a certain lead attribute such as deal size, or it could be a combination such as, time frame, deal size and budget status. It can also depend on whether data is entered for some attributes or not. For example, at the initial stages, a lead that has a contact number may get 100 points, then if time frame is entered, it may get 200 more points. As a salesperson works on the lead
and enters additional data for the lead, the score is automatically recalculated to reflect the latest truth. By giving a numerical score value for each lead, a salesperson can better prioritize their lead lists and focus on those leads with the greatest score value.

**Lead Assignment Rules: Explained**

You can specify assignment rules and rule sets for assignment of resource candidate objects to the lead work object. Assignment Manager uses rules to evaluate and recommend candidate assignments for your specified lead work objects.

Assignment rules are created using work objects, candidate objects, attributes, and conditions. You can use multiple types of assignment rules and rule sets for assignment of candidate objects, such as rank, qualification status, and resources, to the lead work object. For example, you can specify the assignment rule that assigns resources to sales leads by assigning individual sales resources that meet the rule criteria.

**Assignment Rule Criteria**

An assignment rule can have one or more assignment criteria. For example, all leads lower than a certain deal size, and for a specific product are all assigned to a specific Partner resource. Another example of creating an assignment rule might be if you want all leads that are generated based on a specific campaign, such as a CEO round table discussion event, assigned to a specific salesperson.

**Assignment Rule Considerations**

When designing rules, carefully consider how you want to match the lead candidates to the lead work objects. For example, would you want resources assigned based on their geographic location, or their product knowledge, or their skill level, or a combination of any of these attributes? Do you want to match candidates only, or would you like to match them and score them? In a multiple candidate scenario, do you want to assign all matching candidates or only those who achieve higher than a specific score? These are the basic decisions you should make before creating rules.

**Lead Ownership and Sales Team Resources: Explained**

The market is typically organized into territories that comprise customers and prospects. Marketing is closely aligned with sales, and marketing activities are launched to generate leads and maintain the strength of the sales pipeline.

Resources who access leads have different roles as follows:

- Operations support for an automated process to capture leads, prioritize leads for sales engagement, and distribute the leads to appropriate sales or territory team resources.

- Marketing and the lead qualifier role involves lead monitoring, lead reassignment, and continuous review and adjustment of lead quality.

- Sales and territory teams enable lead qualification, perform follow-up lead activities, and convert leads to opportunities.
This topic explains:

- Lead, Sales, and Territory Resources
- Assignment of Leads to Marketing and Sales Resources
- Sales Resource Role
- Resource Privileges and Access Levels

**Lead, Sales, and Territory Resources**

Sales resources are organized into flexible teams and are associated with the sales territories. These sales territories are then assigned to customers, leads, and opportunities to carry out the sales process. The lead follow-up process includes a lead team comprised of individual sales resources who are predominantly active during the lead qualification stage. A lead is then assigned to the appropriate sales team, and a territory team is created for the lead. All sales resources who are assigned to the territory team can view and follow up the lead.

**Assignment of Leads to Marketing and Sales Resources**

Qualified leads are assigned to a sales team based on sales territories. Unqualified leads are assigned to individual lead qualifiers either manually or based on rules defined in Assignment Manager.

**Sales Resource Role**

The sales resource performs the following activities:

- Review quality leads which are augmented with sales collateral, marketing content, customer contact interactions, and references.
- Qualify and assess the lead quality further with the help of customized assessment templates.
- Use the resource picker to manually select a resource to add to the team.
  - Include a description to indicate what role the resource has on the sales team. Many sales team members can access each lead, and each team member is identified as either an internal (sales force), or an external (channel partner sales force) resource. Each sales team member can be associated with a specific resource role to indicate what capacity the member has on the lead.
- Add additional contacts and products to the lead as the lead moves further down the sales cycle.

**Resource Privileges and Access Levels**

Leads have three levels of access as follows:

<table>
<thead>
<tr>
<th>Access</th>
<th>Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>Read and update the lead and all child objects of the lead. Full access level allows you to update the sales lead team by adding or removing individual resources, or by updating the access level for any member.</td>
</tr>
<tr>
<td>Privilege</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>View Only</td>
<td>View the lead and add lead notes. View the sales account associated with the lead, but no other leads or opportunities associated with the sales account. The View Only privilege also allows you to view most of the lead tabs.</td>
</tr>
<tr>
<td>Edit</td>
<td>Update all data on the lead except for Lead Team membership data and Lead Owner data.</td>
</tr>
</tbody>
</table>

When the lead does not have any owner, you must Accept the lead which makes you the lead owner. Only the lead owner and the management chain of the lead owner can change the lead owner.

Territory team members inherit the access level of the territory. All members of the sales territories assigned to the lead have full access to the lead. Owners of ancestor territories of all sales territories assigned the lead also have full access to the lead.

**Defining Automatic Assignment of Lead Team Resources: Example**

Using a concurrent program, administrators can set up assignment manager to automatically assign lead team members and appropriate sales territories to leads. The concurrent program accepts criteria to determine the batch of leads to be assigned. During the rule-based assignment phase of the batch process, matching candidates are automatically added to the lead team. During the territory-based assignment phase of the batch process, territories are automatically added to the lead territory team.

**Scenario**

The Automotive sector uses geography data and overall lead quality to periodically assign leads to automotive dealers. Leads can be automatically scored and ranked before being assigned a resource. Leads can be automatically assigned through Assignment Manager by associating assignment criteria to assignment rules. Leads can be assigned based on Lead Source, Geography, Lead Score, Named Accounts, (such as the top 20), Industry, and Product. For cases where partners need to engage with customers, Assignment Manager can automatically assign partner leads to internal channel managers to ensure a timely sales follow up.

**Manual Setup Assignment Manager Tasks**

The following setup tasks are performed manually through the Assignment Manager user interface:

- Define lead distribution rules
- Define partner matching rules
- Define what makes a quality lead for sales follow up
- Setup lead qualification rules
- Setup lead classification rules to assign leads to specific sales channel
• Setup lead routing rules to route leads to sales resources in a selected sales channel

• Setup lead distribution rules to assign leads to a specific salesperson or partner sales territories

Analysis

If the lead needs rule-based assignment, then rule sets are used. If the lead needs territory-based assignment only, then rule sets are not used. During lead import, either ranking or scoring is used as the default qualification criteria. However, lead scoring can be followed by lead ranking where score is used to determine lead rank, and both Lead Score and Lead Rank can also be used as criteria for lead assignment.

Provided that the following tasks are set to occur one after another by scheduling in sequence, then, when invoked in batch mode, Assignment Manager will automatically:

• Assign leads to sales channel

• Route leads to sales organizations and sales territories

• Assign leads to individual salespersons or partners

You can also manually assign leads to specific internal or external resources. The lead processing activity user interface supports the selection of the type of job, selection of rule set, filtering criteria for selecting leads, and scheduling options.

Sales Team Examples

A lead sales team comprises assigned territories and individual team members. The following examples illustrate some of the features available for the lead sales team:

• Automate assignment of territories to lead territory team

• Automate assignment of individual salespersons to sales team

• Add ad hoc members to sales team

• Update access rights based on the resource

• Change the lead owner

Automate assignment of territories to lead territory team

A lead exists with XYZ Company to purchase 50 large wind generator units in several Western Region states. To ensure that Western Region salespersons get assigned to the lead, the administrator has set up Assignment Manager to automatically add the Western Region territory to the lead territory team.

Sales departments arrange the sales force based on sales territories. Sales resources are organized into flexible teams and are associated with the sales territories. These sales territories are then assigned to customers, leads, and opportunities to carry out the sales process. A territory is the range of responsibility of salespersons over a set of sales accounts. Territories are assigned
to sales accounts when the sales accounts are created. The lead sales team comprises the assigned territories and special resources who are manually assigned to the team on an ad hoc basis.

Automate assignment of individual salespersons to sales team

The lead sales team for XYZ company wants to add a support person to the lead. Typically, support people are not part of any sales territory. There is a rule set group which assigns support team members as individual resources based on rules which match the lead product with the specific support team members.

Add ad hoc members to sales team

Generally, sales team resources are automatically assigned to leads based on configured assignment rules. The following scenarios provide examples for when you may want to manually add additional team members to assist with the lead.

The lead owner, who has full access to the XYZ lead, wants to add one of his company’s contractual experts to his team to help pursue the lead. The lead owner manually invokes a resource picker and selects the ad hoc resource that he wants to add to his team.

When pursuing a lead for an insurance policy, the customer contact requests a unique and complex combination of policy components that require an expert in the company to review. The lead owner adds the expert resource to the lead with full access so they can update the lead with valid combinations of products and services, and, if required, add more team members to the team.

Finally, a salesperson is pursuing a lead that requires the export of products outside the country. The salesperson wants to ensure there are no legal issues with exporting the products and adds a member of their company’s legal counsel to the lead to review the details before contacting the customer again.

Update access rights based on the resource

When a resource is initially added to the lead sales team through rules-based assignment, a profile option setting determines the member’s default access level. Resources in the management hierarchy of a newly added team member have the same level of access to the sales leads as the team member.

All members of the sales territories assigned to the lead have full access to the lead. Owners of ancestor territories of all sales territories assigned the lead also have full access to the lead.

Change the lead owner

Only the lead owner or the resources in the management hierarchy of the lead owner can change the lead owner.

Lead Scoring: Example

A lead score is used only when the lead is easily quantified. The score may be used to calculate the lead rank. You can schedule when and how often to process lead scoring through Assignment Manager.
How Lead Score is Determined

Lead scoring capability requires the rules engine to determine a numerical score based on the value of the lead attribute participating in the rule. For example, consider the following rule.

If a lead contact is a high level Executive, then add a score of 100. If the lead contact is an Operations Manager, then add a score of 50.

When this rule evaluates, it determines the score of the lead based on job title of the lead contact. Once the rules engine evaluates all such rules, the end result of the scoring process is the aggregate score, which is then recorded in the Lead Score attribute.

Data Points for Lead Scoring

The following data points form part of the overall score evaluation:

- All data included on the lead and Primary product
- Lead source data such as campaign attributes
- All customer profile data including industry
- All contact profile data

Lead Ranking: Explained

During lead management setup, you can predefine criteria to rank leads and automate the assignment of leads to the appropriate resource in your organization.

You can define lead rank to categorize leads into buckets such as Hot, Warm, or Cool.

The following data points are available to help evaluate lead ranking rules:

- All data included on the lead and lead primary product data
- All customer profile data including industry and customer size classifications
- All contact profile data
- All lead qualification data

Although lead rank and lead score are not the same, they serve a very similar purpose. Score is often used only when the lead is easily quantified, and then may be used to calculate the lead rank.

Setting Up Lead Ranking

The following summarizes the lead ranking process for predefined lead work objects, rank candidates and associations:

- Select predefined lead work object in Assignment Manager.
- Select predefined ranking candidate object in Assignment Manager.
• Associate ranking candidate objects to lead work object.

Note

There is no seeded mapping for the rank candidate object in the Assignment Objects. Mapping is only for territory-based assignment.

• Use the predefined objects during the creation of assignment rules (rule-based assignment).

• Assign rules to determine the appropriate classification of a work object that will provide a rank value for the lead.

Creating Ranking Rules

Create a rule set with a rule set type of Classification Rule. Set the work object as lead and the candidate object as lead rank. Create a rule with conditions that match the attribute settings you want a lead to have in order to give it a rank value. For example, you might choose the "Decision Maker Identified" attribute name and then select the equal (=) operator. Finally, enter the value of True. Create your remaining conditions, and then enter the action for your rule, such as "Return the candidate value as Hot".

Calculating Lead Rank Based On Score

You can schedule when and how often to process lead ranking where the lead rank value or score is calculated and displayed on screen. Ranking rules, used by Assignment manager, determine what rank to assign to a lead. You can also create ranking rules that use the lead score as the criteria to name each range of scores with a specific rank. For example, if lead score is between a value range of 0 and 39, create a rule to rank the lead as low priority. If the lead score is between a value of 40 and a value of 60, then you might want to create a ranking rule that assigns a medium rank to the lead. Assignment Manager passes the rank value to the Lead Management application and is presented in the user interface as a list. You can choose to override the value by select a different predefined rank code or value from the list. You can also choose the Rank option from the Actions menu to automatically assign a rank for your selected lead.

Lead Ranking: Examples

You can define lead rank to categorize leads into buckets such as Hot, Warm, or Cool leads. Such categorization of leads enables a salesperson to quickly prioritize leads for follow-up activities.

Creating a Lead Rank Rule

Your organization wants to assign a rank of Hot to those leads that have a set time frame and a decision maker identified. Assignment Manager for leads has predefined the lead work object and lead candidate object and you will set up an assignment rule to determine the appropriate classification to apply a rank to all leads for your organization.

1. From the Manage Sales Lead Assignment Rules page, create a new rule set.
2. Select the classification rule type, Sales Lead work object, and Lead Rank as the candidate object.

3. From the Associated Rule Set Groups tab, create a rule set group and name it Ranking. You can search and select another rule set group (if one exists), to associate to the classification rule type.

4. Set the conditions for each rule that the rules engine checks during assignment processing. For example, enter the following rule conditions:
   - Object: Sales Lead Work Object
   - Attribute: Time Frame
   - Operator: Equals
   - Value: Three months

5. Click Add Row icon in Conditions and enter the following details:
   - Object: Sales Lead Work Object
   - Attribute: Decision Maker Identified
   - Operator: Equals
   - Value: True
   - Action: Return the candidate value as Hot

6. Click Save and Close.

Assignment Manager will:
   - Find the matching leads
   - Execute the rules
   - Assign the rank value
   - Pass the rank value onto the lead

**Lead Qualification: Explained**

Qualifying leads is an important first step in bringing the sales lead to a conclusion. At the end of the lead qualification process, the lead can either be classified as a qualified lead which is ready for conversion to an opportunity, or can be retired if purchase interest for the lead cannot be validated. Lead qualification process can either be performed by internal marketing or internal sales groups.

What constitutes a qualified lead varies from company to company.

**Basic Lead Qualification**

In some companies, the basic lead qualification data, including customer budget status and time frame, gathered by lead qualifiers is considered in the scheduled
automated process that determines the lead qualification status value based on rules.

**Additional Lead Qualification**

In other companies, a lead qualification questionnaire score is a factor used by the lead qualifier or salesperson to decide to manually set the lead to a Qualified status using the lead actions menu. Your application administrator assigns the questionnaire to your Lead Qualification Template profile. The answers entered are assessed using a weighted scoring model with instant feedback available via a status bar.

**Lead Status: How it Automatically Gets Set**

Lead quality is assessed as soon as a lead is generated. Lead quality of a newly created lead is mainly determined based on the characteristics of the customer contact on the lead, the type of response which caused the lead to get generated, and the nature of the campaign. Lead quality is further assessed based on the added qualification data such as customer need, urgency or time frame for the project, and whether the customer has set aside a budget for this product.

**Settings That Affect Lead Qualification Status**

Leads can get their qualification status from:

- **Assignment Manager rules**
  
  Rules-based leads qualification process helps standardize the lead qualification process. Based on the positive results to conditional rules, the value of the Lead Status attribute is set to Qualified. For example, a rule can be defined to update the lead as qualified if the customer’s budget status is approved, the project time frame is three months, a decision maker is identified, and the response type is that they attended an event.

- **A specified value in the campaign lead generation stage**
  
  A multistage campaign design can include lead generation stages. Lead options include the ability to designate a value for the qualification status. When the campaign is executed, the leads will be created with the value provided.

- **An imported value**
  
  Leads imported through file import can include a designated lead qualification status.

**How Lead Qualification Status Is Calculated**

In some companies, the lead qualification data gathered by lead qualifiers is considered in the scheduled automated process that calculates lead score or lead rank as well as assigning sales team territories. For such companies, a simple rule to move leads to a Qualified status when the lead score reaches a specific threshold is sufficient.
FAQs for Configure Assignment Manager for Lead Processing

What's the difference between rule-based and mapping-based assignment?

For rule-based assignment, you use the rules editor to create rules that the assignment engine uses to make candidate assignments.

For mapping-based assignment, you create work object to candidate object mappings during assignment object creation that are used to make candidate assignments.

In a rule-based assignment scenario, you create rules with conditions that need to be met in order to make the appropriate candidate assignments. For example, a rule is created with the condition that the candidate object (resource) has a product skill rating that is greater than or equal to intermediate in order to be matched to the work object and be assigned.

In a mapping-based assignment scenario, you typically create attribute to attribute mappings between a work object and a candidate object in order to make the appropriate candidate assignment. For example, a mapping is created between the work object of opportunity revenue line and the candidate object of territory. The mapping is set up to map the attribute of location on the opportunity (the parent of the revenue line) to the geography attribute on the territory; so any territories where the geography value matches the location of the parent opportunity will be matched and assigned to that opportunity revenue line.

What's the difference between literal, dimension, and attribute mapping?

Attribute Mapping: This mapping enables you to compare and match attribute values between a work object attribute and a candidate attribute. When the candidate attribute matches the work object attribute the candidate is selected. Attribute mappings should be used when the work object and candidate object attributes in the comparison are non-dimensional attributes. For example, consider a lead work object with a program ID attribute and the territory object with program ID attribute. The selection criteria is: select Sales Territories where Sales Territory.TerritoryLocation equals Sales Opportunity.OpportunityLocation The assignment engine will use this mapping data to construct a query on the candidate object that is equivalent to the selection criteria. When creating the mapping, use the Function Code field to specify a unique identifier for the dimension. This identifier will be passed to the translation function, in case the same function is used for multiple dimensions.

Literal Mapping: Literal Mapping is used almost exclusively to filter the candidate objects. This forma of mapping enables the comparison of candidate attributes against a specific value chosen by the user. The assignment engine will compare the mapped candidate object attribute against the specified literal value. For example: Select the Territory Candidate object that has the attribute TerrStatusCode that equals the value FINALIZED.
Dimension Mapping: Dimension mapping should be used when the work object and candidate object attributes in the comparison are dimension attributes, such as Geography, Product, or Account. When creating the mapping, use the Function Code field to specify a unique identifier for the dimension. This identifier will be passed to the translation function, in case the same function is used for multiple dimensions.

**What happens if I mark an assignment object or one of its attributes as inactive?**

When the assignment object inactive box is checked the selected work or candidate assignment object is not available for assignment processing. When the assignment attribute inactive box is checked the selected work or candidate object attribute is not available for assignment processing.

**Note**

The object or attribute cannot be set to inactive if there is a mapping or rule defined using the object or attribute.

**How can I identify a Classification candidate object?**

Enter the word Classification in the Application Module field. This will create a candidate object that you can use when setting up classification-type rules; for example, rules that qualify or rank leads.

**How can I exclude a candidate from an assignment?**

You can exclude candidates during the assignment process by navigating to the Candidates tab. Scrolling down the section Exclude Candidates. Enter the View Object that controls which candidates are to be excluded and enter the selection criteria for up to three key fields.

**Note**

The application using Assignment Manager needs to support a View Object that controls which candidates are to be excluded in order to use this feature.

**Configure Assessment Reference Data for Sales Leads**

**Sales Lead Qualification Template: Explained**

Lead qualification templates enable a uniform lead qualification process across leads.

Using lead qualification templates, you can:

- Define Lead Qualification Templates
- Assign Lead Qualification Templates
Gather Lead Quality Information

Define Lead Qualification Templates

Qualification templates are collections of questions and answers that can be weighted and scored to evaluate the quality of gathered information about the sales lead. The templates are used to define consistent and specific qualification criteria for leads.

Assign Lead Qualification Templates

Once the qualification template is defined, the template is assigned to the Lead Qualification Template profile available from the Manage Sales Lead Administrator Profile Values implementation task. If defining more than one qualification template, assign templates at the profile Site level.

Gather Lead Quality Information

Along with other key qualification fields available in the lead, the qualification template questions display on the lead Qualification tab if you have enabled the Advanced Lead Qualification profile option. Lead qualification is typically conducted through phone conversations. As responses are entered for qualification questions, the qualification template’s weighted score is calculated and a progress bar provides immediate rating and feedback.

Defining a Sales Lead Qualification Template: Example

This example illustrates defining a lead qualification template to assess the basic quality of a lead.

Scenario

XYZ Company sells alternative energy solutions to small businesses. They employ a group of people that qualify the basic information about a lead plus gather information needed by the sales team to prepare for follow-up sales calls.

Unqualified leads are generated from various sources and include when customers have requested more information from their website. Before creating the qualification template, XYZ Company evaluates the following:

- To verify the customer’s intent in requesting more information and to ensure a consistent and thorough communication with the customer, a set of questions each lead qualifier is expected to ask is compiled.

- Once the initial information is gathered, only leads with a greater potential for a sale are transferred to the sales team. To assist the lead qualifier in identifying the leads to transfer to sales, the qualification information is categorized into four basic levels ranging from “very little information was obtained from the customer” to “a high potential of the customer working on an alternative energy project”. Question responses are determined for each question including responses such as “Not applicable” and “Information not available” to ensure the full range of possible responses.

- Question responses and questions are analyzed and modeled to effectively place the qualification template weighted score ranges into
Define Lead Management

the four categories. For example, a budget that expires less than ten days may score low if the average sales cycle is greater than 10 days but any answer to a budgeting question has higher weight in the overall questionnaire than a question about the customer's project team. If a qualification template's weighted score is 90 or above, the customer would be categorized as a high potential for an energy project and should be transferred to a sales resource to pursue.

Defining the Qualification Template

Navigate to Manage Assessment Templates from the Manage Sales Lead Qualification Template implementation task. The following categories of qualification levels are entered as ratings of:

- Little Information Available
- Low Project Potential
- Medium Project Potential
- High Project Potential

Questions about the customer's budget, time frame, decision maker, and project drive the sale potential and resulting lead rank. These questions are not included in the qualification template since the lead qualifier will enter those directly in the lead user interface.

Analysis

Since the qualification template will be used to better evaluate the customer's project and prepare the sales team to progress on the lead if the lead is qualified. A set of questions and responses are entered to qualify the time frame including what phase the customer's implementation project is in. To gather information to meet export policies, questions are added about implementation projects outside of the country. Questions are added to determine if and when a customer's budget will expire. To identify competition, questions and responses are entered about competing suppliers or in-house solutions and satisfaction levels. A free-form response is allowed to enable the lead qualifier to capture specific customer comments about their greatest issue they are trying to solve.

Next, the weighed scores ranges are assigned to the four ratings. To provide a visual queue and feedback to the lead qualifier to guide them in their decision to update the lead as qualified, a color and feedback phrase is entered for the four ratings. The color-coded bar, score, and feedback appear at the top of each qualification questionnaire in the lead.

Once the qualification template is defined, the template is assigned to the Lead Qualification Template profile available from the Manage Sales Lead Administrator Profile Values implementation task. Because there is only one business unit for XYZ Company and only one qualification template, the template is assigned at the profile Site level.

As XYZ Company continues to grow into a global company with a broader set of business units and products, Qualification templates are created in different languages and assigned to business unit sets. The appropriate qualification template is assigned to corresponding lead qualification users through the Lead Qualification Template profile.
Note

The Advanced Lead Qualification Enabled profile option must be set to enable you specify the template name to display the collection of questions, possible answers, and questionnaire feedback in the Edit Lead user interface.

Assessment Templates: Points to Consider

Assessment templates let you analyze the health of a business object, such as a lead or an opportunity, and suggest appropriate next steps based on its diagnosis. To best plan and create assessment templates, you should consider the following points:

- Ratings
- Questions, Question Groups, and Question Weights
- Responses and Scores
- Associated Task Templates

Ratings

A rating is a textual qualification such as Excellent. There are three delivered ratings in the assessment template: Excellent, Average, and Poor. Ratings provide a metric other than a numerical score for qualifying the outcome of an assessment. Ratings are created at the beginning of the assessment template creation process. They are later applied to possible responses to questions in the template, which associates each rating with a score. An appropriate feedback will be displayed to you based on the completed assessment score once you submit an assessment. When setting up ratings and applying them to possible responses, it is important to remember that they and their associated feedback text will eventually display as part of the overall assessed health of a business object.

Questions, Question Groups, and Question Weights

Questions are the main components of an assessment template. They are written such that they aid in systematically determining the health of a business object, and they are grouped into logical collections called Question Groups. Each question in the template is assigned a question weight, expressed as a percentage, which is the relative importance of the question within the template. When an assessment template is used to perform an assessment, a question’s weight is multiplied by the response score given for the question to produce a weighted score for that question. When setting up questions, question groups, and question weights, it is important to carefully analyze which factors determine the health of a particular business object (like a lead or an opportunity) in your organization. Use those factors to create your question groups; and then, for example, write three to five questions per group that are weighted according to your analysis. There is no limit to the number of questions that can be in a question group, but each question group must have at least one question.
Responses and Scores

Responses are attached to questions in the template. Each question should have at least two responses, unless it’s a free-form only question. More than one response can be tied to the same rating but, between all of its responses, each question should accommodate at least two ratings, unless it’s a free-form only question. For example, if your ratings are Excellent, Average, or Poor you may, for each question, include two responses that correspond to at least one of those ratings, such as average. There must be enough responses to cover at least two of the ratings such as Excellent and Average. You assign a score to each response for a question, and the application normalizes the score based on a standard scoring scale. When an assessment template is used to perform an assessment, a question’s weight is multiplied by the normalized score of the response given for the question to produce a weighted score for that response. When adding responses to questions, ensure that the scores and ratings you assign to each response correlate. In other words, the higher the score you assign to the response, the higher the rating should be so that you have a strong quantitative relationship between the two. Also note that you can allow free-form responses for one or more questions in the template, but free-form responses are never scored.

Associated Task Templates

A task template is an instruction to generate a group of related activities. You can associate task templates with an assessment template in order to recommend tasks that should be performed after an assessment has been done for a business object. When you associate task templates with an assessment template, you can indicate a score range for each task template, and based on the total score of any assessment that uses your template, one or more task templates will be recommended as follow-up activities. In order for a task template to be available to associate with an assessment template, it must be assigned to the same business object type as that assigned to the assessment template, and it must have a subtype of Assessment. Ensure that you have set up task templates correctly before attempting to associate them to assessment templates.

Assessment Template Status Codes: Explained

Throughout the life of an assessment template, it can be assigned several different status codes.

These status codes control the actions you are allowed to make against an assessment template.

- In Progress
- Active
- Retired

In Progress

This is the initial status of an assessment template. When an assessment template is at this status, you can edit any part of it. This is the only status at which you
can delete a template. If the template is not deleted, it moves to the Active status next.

Active

This is the status assigned when the assessment template has been deployed for general usage. When an assessment template is at this status, you can make only minor textual edits to it, including, but not limited to, template description, question text correction, question sequencing change, response description, and score range feedback. From this status, you can move the template to Retired; you cannot delete it.

Retired

When an assessment template is at this status, it is no longer available for general usage. You cannot edit any part of it, and you cannot move it to any other status; however, it can still be copied. Active templates that are deleted revert to this status.

Assessment Template Score Range: How It's Calculated

The application calculates the score range for an assessment template using the question weights and the ratings and scores assigned to the possible responses for all the questions in the template. This topic explains when the score range is calculated and the components that are used in the calculation, so that you can make the best decision regarding the feedback text to apply to each score range. In addition to the automatic score range calculation, a manual method for adjusting score range is also available on the administration UI.

Settings That Affect Score Range

In order for the application to calculate the assessment template score range, you must:

- Apply weights to all template questions.
- Configure ratings and apply them to possible responses for all template questions.
- Apply a score to each of the possible responses for all template questions.

How Score Range Is Calculated

The score ranges for each rating in an assessment template are determined using the lowest and the highest weighted response scores for each question. So for each rating score range, the lower end of the range starts where the previous rating range ended, and the higher end of the range is the sum of the highest weighted scores that can be attained for that rating.

This table displays a simple example of the components used in the score range calculation.
<table>
<thead>
<tr>
<th>Question (Weight)</th>
<th>Response (Normalized Score)</th>
<th>Weighted Score</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the customer win? (20%)</td>
<td>Lower Operating Cost (100)</td>
<td>20</td>
<td>Excellent</td>
</tr>
<tr>
<td>Higher Revenues (80)</td>
<td>16</td>
<td>Average</td>
<td></td>
</tr>
<tr>
<td>Other (53)</td>
<td>11</td>
<td>Average</td>
<td></td>
</tr>
<tr>
<td>Don't Know (27)</td>
<td>5</td>
<td>Poor</td>
<td></td>
</tr>
<tr>
<td>What is our win? (80%)</td>
<td>Reference (60)</td>
<td>48</td>
<td>Average</td>
</tr>
<tr>
<td>Resale (50)</td>
<td>40</td>
<td>Poor</td>
<td></td>
</tr>
<tr>
<td>Partnership (100)</td>
<td>80</td>
<td>Excellent</td>
<td></td>
</tr>
</tbody>
</table>

This table displays the score range calculation based on the components from the first table.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Score Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excellent</td>
<td>65 - 100</td>
</tr>
<tr>
<td>Average</td>
<td>46 - 64</td>
</tr>
<tr>
<td>Poor</td>
<td>0 - 45</td>
</tr>
</tbody>
</table>

Note

If a template administrator does not use a particular rating while assigning ratings to possible responses, this could result in improper score range calculations. To counteract this problem, the score range calculation uses a built-in correction algorithm to ensure proper score ranges. The correction algorithm works like this: For a question where a particular rating is skipped, the low score for the skipped rating is calculated to be equal to the high score of the next lower ranked rating. The high score for the skipped rating is calculated to be equal to the low score of the next higher ranked rating.

Using the ratings displayed in the tables above, if the rating Average is not used for a question’s possible responses, the score range calculation assigns a low score to Average for that question that is equal to the high score of Poor for that question. It also assigns a high score to Average for that question that is equal to the low score of Excellent for that question. This ensures that the overall template score range for Average is calculated to fall between the score ranges for Poor and Excellent.

Assessment Template Components: How They Fit Together

The question weight, response score, and response rating are the assessment template components that fit together to calculate and display the overall assessment score, rating, and feedback text.

A question weight is multiplied by a response score to achieve a weighted score for an assessment template response. The weighted scores for all responses are added together to determine the total assessment score. This score will fall within a precalculated score range that is associated with a response rating and feedback text. Therefore, the score range within which the total assessment score falls determines the rating and feedback text to display for a completed assessment.
**Question Weight**

The question weight is the relative importance of a question within an assessment template, and it is expressed as a percentage. All of the question weights within a template must total to exactly 100. When an assessment template is used to perform an assessment, a question’s weight is multiplied by the score of the response given for the question to produce a weighted score for that response.

**Response Score**

A response score is the score assigned to a possible response to a question in the template. The template administrator sets response scores with no upper or lower bounds, and each score is normalized in order to accurately score an assessment that uses the template. The response scores are normalized by assigning a score of 100 to the highest response score, and then all other responses are assigned a normalized score relative to that highest score.

When an assessment template is used to perform an assessment, the normalized score of the response given for the question is multiplied by the question’s weight to produce a weighted score for that response.
Response Rating

A response rating is the rating assigned to a possible response to a question in the template. A rating is a textual qualification like Excellent or Poor that provides a metric other than a numerical score for qualifying the outcome of an assessment. A response rating is directly related to a response score, and this relationship should ensure that a higher score will translate to a higher rating.

Early in the template creation process, the administrator configures ratings to assign to responses. The administrator then assigns scores and ratings to responses, and the system calculates score ranges based on those entries. Each rating is assigned to a score range, and the administrator is given the opportunity to apply feedback text to the rating-score range combination.

When an assessment template is used to perform an assessment, the weighted scores from all responses are added to determine the total assessment score. That score will fall somewhere within the calculated score ranges, which then determines which rating is assigned to the assessment and what feedback text to display. The maximum total assessment score is 100.

Assessment Templates and Task Templates: How They Fit Together

One of the steps for creating an assessment template is associating task templates. You would take this step if you want to recommend sets of tasks to be done after an assessment is performed using your template. You associate task templates to ranges of scores in the assessment template, and where the overall assessment score falls within those ranges determines the tasks that are suggested to be performed after the assessment.
Assessment Template
An assessment template is a set of weighted questions and possible responses used to evaluate the health of a business object such as an opportunity or a lead. An assessment template can be associated with one or more task templates that are recommended based on the outcome of an assessment.

Task Template
A task template is an instruction to generate a group of related activities. By marking a task template with a subtype of Assessment, you make that task template available for association with assessment templates. The task template's business object type should be the same as that assigned to the assessment template. When an assessment is performed using an assessment template that has associated task templates, one or more task templates are recommended based on the total score of that assessment and can be used to generate a list of activities to perform.

For example, you can associate a task template called Engage Business Development Manager with your assessment template called Potential for Win-Win. Associate the task template with the score range of 86 to 100, so if an assessment using the assessment template Potential for Win-Win scores within that range, the application recommends the Engage Business Development Manager task template and a list of follow-up activities based on that template can be generated.

FAQs for Configure Assessment Reference Data for Sales Leads

What happens if I include a free-form response for a question?

A score of 0 is assigned for free-form responses. A free-form response option will have no effect on the overall assessment score. The free-form response offers the opportunity to enter a textual response to a question that does not conform to any of the pre-populated responses provided by the assessment template.

What's a Question Group?

A question group is a logical grouping of questions within an assessment template, and it is used strictly as a category header for those questions. Through careful naming of a question group, you can achieve the benefit of providing the user of the template with an approximate idea of the type of questions to expect in each group.

Why am I being asked to enter question weights again?

This step lists all of the assessment template questions in one place, and provides you with the opportunity to edit weights as necessary to ensure that the sum of all weights totals 100.

Configure Notes for Sales Leads

Defining Notes: Points to Consider

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:

- Note Types
• Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn’t exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article Extending CRM Applications: how it works.

Configure Tasks for Sales Leads

Turning a Business Process into a Task Template: Example

This example illustrates how to create a task template that represents a business process.

Scenario

A sales manager wants to create a task template for her department’s client product demonstration process.

Client Product Demonstration Activities

The client product demonstration process occurs regularly. The sales manager does not want to manually create tasks for this process every time it occurs, so she decides to create a task template that includes the business process activities. Each time she repeats the business process, she can use the task template to automatically generate the appropriate tasks that need to be performed.

Analysis

The business process consists of the following activities:

• Book a conference room.
• Create an agenda.
• Confirm the date and time with the client.
• Make arrangements with presenters.
• Deliver product demonstration.
Follow up with client.

Based on the analysis of the business process, the following task template is created:

Task Template Name: Client Product Demonstration

<table>
<thead>
<tr>
<th>Task</th>
<th>Category</th>
<th>Lead Days</th>
<th>Duration Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book conference room</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Create agenda</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Confirm date and time with client</td>
<td>Call</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Schedule presenters</td>
<td>Preparation</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Deliver demonstration</td>
<td>Demonstration</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Follow up with client</td>
<td>Call</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

Defining Tasks: Points to Consider

A task is a unit of work to be completed by one or more people by a specific completion date. When using tasks in your application, you should consider the following points:

- Tasks
  - Task Templates

Tasks

A task is defined with a description, due date and category. Each task has an owner, who oversees or is responsible for the task, and one or more assignees who perform the work. The task can be related to a business object, such as an Opportunity, a customer or, one or more external contacts. Tasks may also have Notes for general information, Attachments for tracking e-mail or project documents and Interactions which record customer communications.

Task Templates

Often a set of Tasks are performed repeatedly for a particular process and to support this administrators can define Task Templates, which represent a group of tasks. These task templates can be invoked by users working on a particular business object, for example a lead, campaign, or an opportunity. The user selects the appropriate task template for the goal they want to achieve and this creates the tasks and associates them with the business object being worked on. This saves the user from creating multiple tasks when an Opportunity reaches a particular sales stage, or the same set of tasks each time a marketing campaign is created.

Note

Extensibility features are available on the Task object. For more details refer to the article Extending CRM Applications : How It Works.

How can I create a task template that is available to associate with assessment templates?

Create the task template with a subtype of Assessment.
## Configure Sales Leads

### Sales Lead Lookups: Explained

Lookups enable quick selection from drop-down menus. Oracle Lead Management lookups are incorporated into its application to speed the process of entering data into forms.

The lookups listed in the following table display the lookup name and its associated values and description by lookup type and in alphabetical order.

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Lookup Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead Access Level</td>
<td>• Edit</td>
<td>Access levels for lead team members for a specific lead.</td>
</tr>
<tr>
<td></td>
<td>• Full</td>
<td>Full access level allows the user to update sales lead team by adding or removing individual resources or by updating the access level for any member.</td>
</tr>
<tr>
<td></td>
<td>• View only</td>
<td></td>
</tr>
<tr>
<td>Lead Assignment Process Type</td>
<td>• Assignment</td>
<td>Lead processing types supported by Assignment Manager.</td>
</tr>
<tr>
<td></td>
<td>• Qualification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Ranking</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Realign with territories</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Scoring</td>
<td></td>
</tr>
<tr>
<td>Lead Registration Approval Status</td>
<td>• Approved</td>
<td>Approval status for leads registered by partners.</td>
</tr>
<tr>
<td></td>
<td>• Pending approval</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rejected</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Submitted</td>
<td></td>
</tr>
<tr>
<td>Lead Registration Type</td>
<td>• Co-sell</td>
<td>Types of leads available for partners.</td>
</tr>
<tr>
<td></td>
<td>• Referral</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Resale</td>
<td></td>
</tr>
<tr>
<td>Lead Acceptance Status</td>
<td>• No</td>
<td>Status that indicates whether or not a lead is accepted by sales departments.</td>
</tr>
<tr>
<td></td>
<td>• Yes</td>
<td></td>
</tr>
<tr>
<td>Lead Assignment Status</td>
<td>• Assigned</td>
<td>Assignment status indicating whether sales team resources are assigned.</td>
</tr>
<tr>
<td></td>
<td>• Reassigned</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rejected</td>
<td></td>
</tr>
<tr>
<td>Lead Qualification Budget Status</td>
<td>• Approved</td>
<td>The approval status of a customer budget. Data used to assess lead qualification status.</td>
</tr>
<tr>
<td></td>
<td>• Pending</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Unknown</td>
<td></td>
</tr>
<tr>
<td>Lead Channel</td>
<td>Lead Reassignment Reason</td>
<td>Lead Registered Status</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------</td>
<td>------------------------</td>
</tr>
</tbody>
</table>
| • Direct mail  
• E-Mail  
• Fax  
• Phone  
• Sales visit  
• Web  
• Wireless message  
• Prediction  
• Model based prediction  
• Rule based prediction | • No activity  
• Other  
• Workload | • No  
• Yes | • Immediate  
• Schedule date  
• Repeats | • Completed  
• Completed with error  
• New  
• In progress  
• Scheduled | • Converted  
• Qualified  
• Retired  
• Unqualified |

Source channel responsible for lead generation.

Lead rank values used as a measure of lead quality and prioritization.

Possible reasons specified for reassigning leads.

Status that indicates whether or not a partner lead is registered.

Scheduling options for lead processing activities.

Progress details indicating lead current and end processing status.

Lead status values based on specific actions performed on a lead. Used to mark the milestones in the lead life cycle.
### Lead Time Frame

- 3 months
- 6 months
- 9 months
- 12 months
- 15 months
- 18 months

Lead cycle duration that usually coincides with a typical sales cycle duration for products and services offered.

### Lead Reject Reason

- Duplicate lead
- Failed to reach contact
- Incorrect data

Possible reasons specified for rejecting leads. Rejected leads can be reassigned or retired.

### Lead Retire Reason

- Duplicate lead
- No purchase interest

Possible reasons for retiring leads. Retired leads are considered closed leads.

### Lead Process Scheduling Operators

- Equal to
- Greater than
- Greater than or equal to
- Less than
- Less than or equal to

Operators used for lead processing scheduling options.

### Recommended Lead Actions Type

- Accept leads
- Create personal campaign

Type of actions to perform on recommended leads.

### Lead Processing Repeat Frequency

- Days
- Months
- Weeks
- Years

The time intervals between lead processing activities.

---

Lookups in applications are used to represent a set of codes and their translated meanings. For example, a product team might store the values ‘Y’ and ‘N’ in a column in a table, but when displaying those values they would want to display “Yes” or “No” (or their translated equivalents) instead. Each set of related codes is identified as a lookup type. There are many different examples of these across Oracle Fusion Applications. You can add lookups and add values to the default lists.

## Sales Lead Profile Options: Explained

Profile options are configurable options that affect application operations. Values defined at the user level take precedence over those at the site level. If a value is not defined at the user level, the site level value is used.

Sales lead profile options are categorized into the following:

- Lead Assignment
- Lead Qualification and Assessment
• Partner Leads
• Lead and Recommended Lead Display in Sales Targeting
• Leads List Views

**Lead Assignment**

The following table lists the profile options that affect the assignment of lead status, rank, score, resources, and territories to the lead. The values provided are used by the Sales Lead Processing Activities for scheduled assignment processing and the rank, score, and reassign actions from the lead. The Configure Assignment Manager for Lead Processing task list is a prerequisite.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment Rule for Qualifying Leads</td>
<td>Rules to evaluate the lead and assign the lead status per rule conditions.</td>
</tr>
<tr>
<td>Assignment Rule for Ranking Leads</td>
<td>Rules to evaluate the lead and assign the lead rank per rule conditions.</td>
</tr>
<tr>
<td>Assignment Rule for Rule-Based Lead Assignment</td>
<td>Rules to evaluate the lead and assign the sale team resources per rule conditions.</td>
</tr>
<tr>
<td>Assignment Rule for Scoring Leads</td>
<td>Rules to evaluate the lead and assign a score per rule conditions.</td>
</tr>
<tr>
<td>Assignment Rule for Territory-Based Lead Assignment</td>
<td>Rules to evaluate the lead and further filter territories derived using territory based assignment per rule conditions.</td>
</tr>
</tbody>
</table>

**Lead Qualification and Assessment**

The following table lists the profile options that affect the qualification and assessment of the lead. The Configure Assessment Reference Data for Leads task list is a prerequisite.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead Assessment Enabled</td>
<td>Enable the display of Assessments, a tab containing the sets of predefined questions and answers collected to assist in evaluating the lead.</td>
</tr>
<tr>
<td>Lead Assessment Template Default</td>
<td>Specify the assessment template questionnaire to default when creating a new lead assessment.</td>
</tr>
<tr>
<td>Advanced Lead Qualification Enabled</td>
<td>Enable advanced lead qualification by specifying the template name to display the collection of questions, possible answers, and questionnaire feedback.</td>
</tr>
<tr>
<td>Lead Qualification Template</td>
<td>Specify the assessment template to use for evaluating and qualifying leads.</td>
</tr>
</tbody>
</table>

**Partner Leads**

The following table lists the profile options that affect sales partner leads.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead Registration Approval Administrator</td>
<td>Select the recipient of notifications when an approver cannot be determined, who is qualified to respond to the notification, and can modify approval configurations.</td>
</tr>
<tr>
<td>Option Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Lead Registration Expiration Date</td>
<td>Specify the number of days, after the lead registration creation date, used to derive the lead registration expiration date.</td>
</tr>
</tbody>
</table>

**Lead and Recommended Lead Display in Sales Targeting**

The following table lists the profile options that affect the number of leads and recommended leads displayed in the leads work areas and sales targeting heat maps. The four top product profiles are evaluated together to group products into a subset used as the basis for sales targeting analysis. The four top sales account profiles are evaluated together to group customer sales accounts into a subset used as the basis for sales targeting analysis.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option Name</td>
<td>Description</td>
</tr>
<tr>
<td>Top Products for Sales Targeting</td>
<td>Specify the maximum number of retrieved products, using the top product dimension attribute minimum and maximum criteria, to display and analyze for potential leads.</td>
</tr>
<tr>
<td>Top Product Dimension Attribute for Sales Targeting</td>
<td>Select the product dimension attribute used to compare corresponding profile minimum and maximum values when retrieving and sorting the top products.</td>
</tr>
<tr>
<td>Top Product Dimension Attribute Maximum for Sales Targeting</td>
<td>Specify the upper limit of the product dimension attribute value, identified in the corresponding profile, used when retrieving the top products.</td>
</tr>
<tr>
<td>Top Product Dimension Attribute Minimum for Sales Targeting</td>
<td>Specify the lower limit of the product dimension attribute value, identified in the corresponding profile, used when retrieving the top products.</td>
</tr>
<tr>
<td>Top Sales Account Attribute Maximum for Sales Targeting</td>
<td>Specify the upper limit of the sales account attribute value, identified in the corresponding profile, used when retrieving the top sales accounts.</td>
</tr>
<tr>
<td>Top Sales Account Attribute Minimum for Sales Targeting</td>
<td>Specify the lower limit of the sales account attribute value, identified in the corresponding profile, used when retrieving the top sales accounts.</td>
</tr>
<tr>
<td>Top Sales Accounts Attribute for Sales Targeting</td>
<td>Select the sales account attribute used to compare corresponding profile minimum and maximum values when retrieving and sorting the top sales accounts.</td>
</tr>
<tr>
<td>Top Sales Accounts for Sales Targeting</td>
<td>Specify the maximum number of retrieved sales accounts, using the top sales account dimension attribute minimum and maximum criteria, to display and analyze for potential leads.</td>
</tr>
</tbody>
</table>

**Leads List Views**

The following table lists the profile option for sales lead List views.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Days to Query Leads</td>
<td>The profile value controls the default of the lead start and end date criteria values when searching leads based on creation date in the leads work areas. The values can be overridden in the search criteria user interface.</td>
</tr>
</tbody>
</table>
Define Integration Points for Segmentation Manager

Marketing Segments and Segment Trees: Highlights

Segmentation is the practice of dividing a customer base into groups that are similar in specific ways such as demographics or past purchases. Marketers use segmentation to target groups of customers and allocate marketing resources effectively. Segments and segment trees are selected in marketing campaigns to receive marketing communications and to generate leads. Marketing Segmentation supports the creation and management of segments and segment trees.

Manage Marketing Segments

• For information about marketing segments and segment trees, see the Using Marketing Segments and Segment Trees chapter in the Marketing Segmentation Guide.

Manage Marketing Standard URLs

How Standard URLs and Marketing Content Fit Together

When creating marketing content, you can insert various types of URLs. Standard URLs are one type that you predefine and that can be used across many campaigns. Additional URLs are common response forms predefined by Oracle and ad hoc URLs that can be defined while creating content, if a predefined URL does not already exist. An Ad Hoc URL is not reusable.

Standard URLs

Navigate to the Manage Standard URLs task in the Functional Setup Manager application to define Standard URLs that can be easily selected when creating marketing content.

Enable tracking if you want to track e-mail recipient clicks. If enabled, every time a contact clicks a link in a campaign e-mail, the click is recorded as a response, together with details pertaining to the URL.
A dynamic URL is not a URL that recipients will click, but used instead as a source for content to be automatically inserted when the e-mail is sent. Consequently, tracking does not apply to dynamic URLs.

Managing URL Categories

The Marketing URL Category lookup provides the choices for Standard URL categories. Your application administrator can add categories by navigating to the Manage Marketing Standard Lookups task in the Functional Setup Manager application.

Marketing Content

To include these predefined URLs in your marketing content, select the Element toolbar button and then select Standard URL or Dynamic URL. This feature is available for content created or edited in Treatments, Treatment Templates, Mini Campaigns, and Sales Campaigns.

Review Diagnostics for Marketing

Marketing Source Codes: Explained

A marketing source code is a unique identifier that represents a campaign or marketing activity and is used to track marketing effectiveness.

Source Codes

Marketing source codes are automatically generated for every campaign and marketing activity. In a multistage campaign, you can override the generated source code value with your own value at the campaign level and for stages of type Event.

To view a listing of source codes and associated campaign components, navigate to the Review Marketing Source Codes task in Functional Setup Manager. Note that a source code can be listed more than once in this page to support querying on specific promotions and coupons. Consider the following scenario:

• Two segments are defined, Gold Customers and Silver Customers, representing the two highest levels of preferred customers.

• Two treatments are defined. One intended for Gold Customers that includes a promotion with two coupons. Another intended for Silver Customers that includes a promotion with one coupon.

• A campaign is created for the Summer product catalog. A stage is designed with a fixed date and does not repeat. Each segment is allocated to the corresponding treatment in the stage design.

The source codes for the stage are generated based on the combination of stage instance, audience, and treatment. Since the Gold Customer Product Promotion treatment has two coupons, the source code that represents the stage instance, audience, and treatment combination is listed twice.

The following table illustrates the source codes displayed in the Review Marketing Source Codes page for this scenario.
Using this same example, you can determine all campaign activities that include the Limited Time Discount by using **Query by Example** and entering the coupon name in the field provided above the **Coupon** column.

**Associating Source Codes to Responses and Leads**

Source codes associated with responses and leads provide a reference to marketing campaign activities for tracking campaign effectiveness.

You can provide a source code when creating, editing, or importing a marketing response. A response generated as a result of an e-mail campaign trackable URL or an e-mail bounce inherits the source code derived from the originating e-mail marketing campaign.

A source code can be associated with leads when designing a lead generation stage in a multistage campaign, when manually creating or editing a lead, or when importing a lead. A lead created from a converted response will inherit the source code from the response.

**Define Marketing Promotions**

**Marketing Promotions and Coupons: Explained**

Promotion names and associated coupon codes provide tracking for offers extended to customers.

This topic includes the following:

- Promotions, Coupons, and Marketing Treatments
- Promotions and Mini Campaigns
- Promotions, Coupons, and Source Codes
- Promotions, Coupons, and Marketing Responses
Promotions, Coupons, and Marketing Treatments

You can add promotion names and selective coupon codes to a treatment, where you design the promotional message and coupon content. You allocate the treatment to specific customer segments or lists of contacts when designing your multistage campaign. You can design one treatment per promotion, per coupon, or combinations of promotions and coupons. You can also assign promotions and coupons to more than one treatment.

Promotions and Mini Campaigns

You can add promotion names to your mini campaign, where you design the promotional message and coupon content to be delivered to the customer segments and lists of contacts selected for the campaign. You can design one mini campaign per promotion or multiple promotions, all coupons for the selected promotion are automatically included. You can also design many mini campaigns using the same promotion.

Promotions, Coupons, and Source Codes

A marketing source code is a unique identifier that represents a campaign or marketing activity and is used to track marketing effectiveness. While a source code does not represent individual promotions or coupons, it does represent the campaign stage instance, audience segment or list, and treatment combination of a multistage campaign, and the instance, campaign content, and audience segment or list combination of a mini campaign.

To view a listing of source codes and associated campaign components, navigate to the Review Marketing Source Codes task in Functional Setup Manager. Note that a source code can be listed more than once in this page to support querying on the related promotions and coupons.

Promotions, Coupons, and Marketing Responses

You can provide a source code when creating, editing, or importing a marketing response. A response generated as a result of an e-mail campaign trackable URL or an e-mail bounce inherits the source code derived from the originating e-mail marketing campaign.

By providing a source code on the marketing response, associated promotions and coupons are also added as informative information on the response. You can search responses by promotion name and coupon code.

Define Campaign and Response Notes for Marketing

Defining Notes: Points to Consider

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:
• Note Types

• Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn't exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article Extending CRM Applications: how it works.

Define Campaign Tasks for Marketing

Turning a Business Process into a Task Template: Example

This example illustrates how to create a task template that represents a business process.

Scenario

A sales manager wants to create a task template for her department's client product demonstration process.

Client Product Demonstration Activities

The client product demonstration process occurs regularly. The sales manager does not want to manually create tasks for this process every time it occurs, so she decides to create a task template that includes the business process activities.
Each time she repeats the business process, she can use the task template to automatically generate the appropriate tasks that need to be performed.

**Analysis**

The business process consists of the following activities:

- Book a conference room.
- Create an agenda.
- Confirm the date and time with the client.
- Make arrangements with presenters.
- Deliver product demonstration.
- Follow up with client.

Based on the analysis of the business process, the following task template is created:

**Task Template Name: Client Product Demonstration**

<table>
<thead>
<tr>
<th>Task</th>
<th>Category</th>
<th>Lead Days</th>
<th>Duration Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book conference room</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Create agenda</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Confirm date and time with client</td>
<td>Call</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Schedule presenters</td>
<td>Preparation</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Deliver demonstration</td>
<td>Demonstration</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Follow up with client</td>
<td>Call</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

**Defining Tasks: Points to Consider**

A task is a unit of work to be completed by one or more people by a specific completion date. When using tasks in your application, you should consider the following points:

- Tasks
- Task Templates

**Tasks**

A task is defined with a description, due date and category. Each task has an owner, who oversees or is responsible for the task, and one or more assignees who perform the work. The task can be related to a business object, such as an Opportunity, a customer or, one or more external contacts. Tasks may also have Notes for general information, Attachments for tracking e-mail or project documents and Interactions which record customer communications.
Task Templates

Often a set of Tasks are performed repeatedly for a particular process and to support this administrators can define Task Templates, which represent a group of tasks. These task templates can be invoked by users working on a particular business object, for example a lead, campaign, or an opportunity. The user selects the appropriate task template for the goal they want to achieve and this creates the tasks and associates them with the business object being worked on. This saves the user from creating multiple tasks when an Opportunity reaches a particular sales stage, or the same set of tasks each time a marketing campaign is created.

Note

Extensibility features are available on the Task object. For more details refer to the article Extending CRM Applications: How It Works.

How can I create a task template that is available to associate with assessment templates?

Create the task template with a subtype of Assessment.

Define Marketing Templates: Manage Marketing Campaign Templates

Turning a Business Process into a Task Template: Example

This example illustrates how to create a task template that represents a business process.

Scenario

A sales manager wants to create a task template for her department’s client product demonstration process.

Client Product Demonstration Activities

The client product demonstration process occurs regularly. The sales manager does not want to manually create tasks for this process every time it occurs, so she decides to create a task template that includes the business process activities. Each time she repeats the business process, she can use the task template to automatically generate the appropriate tasks that need to be performed.

Analysis

The business process consists of the following activities:

- Book a conference room.
- Create an agenda.
- Confirm the date and time with the client.
- Make arrangements with presenters.
- Deliver product demonstration.
- Follow up with client.

Based on the analysis of the business process, the following task template is created:

**Task Template Name: Client Product Demonstration**

<table>
<thead>
<tr>
<th>Task</th>
<th>Category</th>
<th>Lead Days</th>
<th>Duration Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book conference room</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Create agenda</td>
<td>Preparation</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Confirm date and time with client</td>
<td>Call</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Schedule presenters</td>
<td>Preparation</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Deliver demonstration</td>
<td>Demonstration</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Follow up with client</td>
<td>Call</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

**Campaign Stage Design Elements: Explained**

The campaign design interface provides the components you use to create stages and specify audiences, treatments, allocation, schedule, event details, additional options, and lead options. The elements available to specify for each stage are determined by the stage type you select.

<table>
<thead>
<tr>
<th>Element</th>
<th>Interaction Stage</th>
<th>Event Stage</th>
<th>Advertising Stage</th>
<th>Lead Generation Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience</td>
<td>Yes</td>
<td></td>
<td></td>
<td>Segments and Segment Trees only</td>
</tr>
<tr>
<td>Treatments</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Allocations</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Schedule</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Event Details</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additional Options</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Lead Options</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

The standard design elements are described in the sections that follow.

**Note**

Since this campaign design functionality is extensible, your enterprise may have customized design elements that are different from the standard set. For example, your enterprise may add additional attributes to the Summary tab, or may add an Additional Attributes tab to capture more event attributes.
**Audience**

The audience collectively describes all the contacts you want to target in the campaign. Only Interaction and Lead Generation stages have audiences. For an Interaction stage, the audience can include any combination of segment trees, segments and lists. Once you launch the campaign, lists are expanded to show individual contacts. For a Lead Generation stage, the audience can include segment trees and segments, but not lists.

The *Quick Add Audience* list provides convenient access to the most widely used segments and lists. Because users can modify the set of segments and lists that appears in the Quick Add list, the list reflects changes made by you and also by other users.

**Treatments**

Treatment is a marketing term describing the content that is to be delivered to the target audience of a marketing campaign. Applicable to Interaction and Advertising stages, you can add any number of treatments from the stage action menu or from the Quick Add list. You and other users can modify the Quick Add Treatment list in the same manner as the Quick Add Audience list. If necessary, you can create new treatments to include in your campaign.

**Allocations**

Allocation describes the process of relating the selected audience members to the corresponding treatments, to determine what content each contact will receive.

Applicable to Interaction stages, the two available modes of allocation are *Simple* and *Advanced*.

- In Simple mode, you can only allocate 100 percent of the audience element, whether segment or list, to a treatment.

- In Advanced mode, you can allocate by count or by percentage, and you can allocate a specific count or percentage of audience members to a control group. The control group is a random sampling of contacts who will not receive the campaign marketing material. You can evaluate the effectiveness of the campaign by comparing responses from control group members to responses from contacts who received the marketing material. In order to have your supplier process the full volume of transactions, you allocate more than 100 percent of the audience. For example, if your supplier is to send 5000 e-mails, and you want a control group of 500, you would allocate 5500 audience members and specify the control group size. When the stage executes, 5500 audience members will be loaded, but 500 will be identified as control group members, and excluded from the launch.

**Schedule**

Applicable to Interaction stages, the schedule shows the start and end times of each system task that will be executed as part of the Load, List Export and
Launch processes. By default, the Load, List Export and Launch processes run consecutively, each immediately following completion of its predecessor, on the date scheduled for the stage.

Optionally, you can enable manual tasks, which can be run by customized BPEL process. You can specify the duration of each process, and can request notification to the owner when it has completed. The dates of all subsequent steps are adjusted automatically to allow for the duration of added processes.

**Event Details**

Applicable to Event stages, you can choose from the list of event types, choose the venue, and optionally, add attachments.

**Additional Options**

Applicable to Interaction and Lead Generation stages, this page provides additional choices for loading audience segments and for lead generation processing.

**Lead Options**

For a Lead Generation stage, once you have added audience segments and trees, select the audiences from which leads are to be created. For each audience, select the source code, so that the resultant leads can be tracked back to the corresponding marketing activity.

**Repeating Campaign Stages: Points to Consider**

Within a single campaign, you can define stages that repeat on a daily, weekly or monthly basis for the duration you specify, and can funnel customers through stages that repeat at different intervals.

You can create and manage repeating stages with varying degrees of complexity, such as the following:

- Repeating a single stage
- Repeating two stages with simple funneling
- Repeating two stages with complex funneling
- Pausing a campaign with repeating stages

**Repeating a Single Stage**

You can create a very simple campaign to target customers who have bought a subscription to your ABC product, to remind them when their subscription is about to expire, and offer a discount if they choose to renew for a longer term.

**Repeating Two Stages with Simple Funneling**

Your campaign has two repeating stages:
• Stage One, which repeats monthly, looks for all customers who buy your ABC product, to send them an e-mail with a link to the online user forum for ABC product. Every time a user finds an answer from the forum, rather than making a support call, your company saves money.

• Stage Two, which repeats quarterly, is funneled from Stage One, and looks for all customers who accessed the online user forum, to send them an offer to purchase one year of support for ABC product at a 50 percent discount.

When the first instance of Stage Two begins, Stage One has already repeated three times. All customers who were targeted in the first three instances of Stage One, and who subsequently visited the online forum, qualify for the discount offer, which is sent to them in the first instance of Stage Two.

When the second instance of Stage Two begins, Stage One has already repeated six times. The first instance of Stage Two has already targeted customers from the first three instances of Stage One. To ensure that no eligible customer gets a duplicate e-mail with the same discount offer, Stage Two must compute a segment which includes only those customers targeted in the last three instances of Stage One.

**Repeating Two Stages with Complex Funneling**

Again, your campaign has two repeating stages:

• Stage One, which repeats quarterly, targets customers who have attended your webinar during the last three months, and sends them a coupon that is valid for one year and which they can redeem online at any time within the year.

• Stage Two, which repeats monthly, is funneled from Stage One to find customers who redeemed their coupon online, plus those who also attended your webinar, and send them invitations to a company event, scheduled near their location, where they can share their experience with other customers from the same area.

Stage Two repeats frequently, to ensure that customers have the opportunity to attend an event soon after they purchase the product and redeem the coupon. The first instance of Stage Two starts soon after the first instance of Stage One, but Stage Two then repeats more frequently than Stage One.

When the third instance of Stage One begins, six instances of Stage Two have already occurred. When the seventh instance of Stage Two begins, it is not enough to target the same customers as in the second instance of Stage One, since there may be some customers who were targeted in the first instance of Stage One but who redeemed their coupons later.

It is therefore better to compute a segment for every instance of Stage Two that:

• includes all customers targeted in all previous instances of Stage One and

• excludes all customers targeted in all previous instances of Stage Two

In this way, all customers who were ever targeted by Stage One are considered for Stage Two, but no customer gets duplicate treatment for Stage Two.
Pausing a Campaign with Repeating Stages

If a campaign is paused, all repeating stages are skipped for the duration of the pause. When the dialog resumes after the pause, any repeating stages execute on their next scheduled occurrence.

Stage Schedule Automatic Tasks: Explained

You can monitor the execution status of all completed and active processes in your campaign, to check that the campaign stages are on track in terms of completion dates and times.

Campaign execution tasks are grouped in three categories:

- **Load**
- **List Export**
- **Launch**

Each category has a mandatory automatic task plus optional, configurable tasks.

**Load**

The load task uses segments or lists to build the audience of campaign members targeted in a particular stage. Every segment has an associated load format, that determines which campaign member attributes are to be loaded. Optionally, you can override this default format by selecting a different load format on the Additional Options tab.

You can tailor the load schedule by enabling optional tasks, either for manual intervention or for a custom action to be executed automatically. Once the load process completes successfully, you can view individual campaign members in the Campaign Execution details for the stage.

**Note**

All segments defined for a stage are loaded at the same time. If a stage has multiple extremely large segments, it will take longer to load. In such cases, you may want to create duplicate stages, and add only one large segment to each iteration, for faster processing.

**List Export**

The list export task takes the data from the load task, generates a text file of recipients, and uploads the file to the Universal Content Manager (UCM). The ID of this text file, together with the ID of the treatment stored in UCM, is made available to the supplier associated with the treatment, in the Launch task.

**Launch**

The launch task does the following:
• If the supplier is Oracle Fusion E-Mail and Web Marketing, the e-mails are delivered using the content and campaign member data uploaded in the List Export phase.

• For third party suppliers, the list of campaign members and content is delivered to them based on their distribution profile settings.

Campaign Stage Additional Options: Explained

Additional options are available for Interaction and Lead Generation campaign stages.

This page has the following fields.

• Purge Cache
• Allowable Error Count
• Original System Reference
• Load Format
• Duplicate Look Back Days

Purge Cache

During the load process, the campaign members for each stage are loaded into the database. For segments, the selected contacts are also stored in the segmentation cache. The segmentation cache is purged automatically 24 hours after loading, so there is no need to select this option if more than 24 hours have elapsed since the last load cycle.

Allowable Error Count

Applies only to Lead Generation stages. Specify the maximum number of errors to allow before stopping the lead import process and the corresponding execution of the stage.

Original System Reference

Applies only to Lead Generation stages. Provide a source reference for the newly imported records.

Load Format

Specify the path and file name to a new mapping that will override the default load format for all segments in the stage.

Duplicate Look Back Days

Applies only to Lead Generation stages. Specify the number of days in the immediate past that the import process will check for duplicates, when importing new leads.
FAQs for Manage Marketing Campaign Templates

What happens if I create a campaign from a template or an existing campaign?

Most planning and design attributes are copied. When creating from a previous campaign or from a template, the following elements are copied:

- Attachments
- Stages
- Audiences associated with stages
- Treatments associated with stages

When creating a campaign from a template, the following additional elements are copied:

- Tasks from any task template associated with the campaign template

Note

It is the task template, rather than the campaign template, that determines which tasks are added to your copy. If the task template has changed since the campaign template was created, the tasks in your copy may not match those in the original.

The following elements are not copied:

- Campaign team
- Notes
- Allocations
- Schedules

What happens to existing campaigns if I update the campaign template?

Existing campaigns that were created from the template are unaffected. Your changes will only affect campaigns created after the template is updated.

Campaign templates provide models to use as a basis when planning and designing multistage campaigns. When you use a template to create a multistage campaign, the template values are copied to the new campaign and the template’s purpose then ends for that campaign.

How can I change the quick add list when designing multistage campaigns?

The Quick Add lists contain collections of favorite treatments and audiences. The lists reflect the most recent changes, as users add and remove items. In the contextual area, click the Edit List link below the Quick Add list you want to change. A window opens, showing all the items available for that list. You can
select one or more items and use the arrow buttons to move your selection into or out of the Quick Add list.

**What's the difference between updating the count and refreshing the allocation?**

Updating the count recalculates the number of members in segments and segment trees. This is useful when the segment rules have been changed, or if you are revisiting the campaign design after an interval where many more contacts may have been added to the segment.

Refreshing the allocation recalculates the matrix where audience members are allocated to treatments. You can refresh your allocation of numbers or of percentages, or you can change between numbers and percentages.

**What's a campaign stage custom action?**

A custom action is a BPEL workflow that your enterprise has created and registered, to extend the automated load capabilities and extend or replace the automated launch capabilities of an interaction stage in a multistage campaign. During campaign design, you can optionally select custom actions when you schedule the execution activities for the stage.

**What happens if I select the notify option for a stage schedule activity?**

When the activity completes, or if it encounters an error condition, an e-mail notification is sent to the owner named for this step in the schedule.

**What happens if I deselect the Enforce Approvals check box after campaign approval has started?**

While approval is pending, a campaign cannot be changed. If you have authority to deselect Enforce Approvals, and decide that you do not want to seek approval for your campaign, you must first withdraw the approval request.

Campaign approval is enabled by a profile option. If approval is enforced, you must obtain approval before the campaign can be activated. You can override the requirement for mandatory approval only if you have the marketing vice president role.

**Define Marketing Templates: Manage Marketing Treatment Templates**

**Marketing Promotions and Coupons: Explained**

Promotion names and associated coupon codes provide tracking for offers extended to customers.
This topic includes the following:

- Promotions, Coupons, and Marketing Treatments
- Promotions and Mini Campaigns
- Promotions, Coupons, and Source Codes
- Promotions, Coupons, and Marketing Responses

**Promotions, Coupons, and Marketing Treatments**

You can add promotion names and selective coupon codes to a treatment, where you design the promotional message and coupon content. You allocate the treatment to specific customer segments or lists of contacts when designing your multistage campaign. You can design one treatment per promotion, per coupon, or combinations of promotions and coupons. You can also assign promotions and coupons to more than one treatment.

**Promotions and Mini Campaigns**

You can add promotion names to your mini campaign, where you design the promotional message and coupon content to be delivered to the customer segments and lists of contacts selected for the campaign. You can design one mini campaign per promotion or multiple promotions, all coupons for the selected promotion are automatically included. You can also design many mini campaigns using the same promotion.

**Promotions, Coupons, and Source Codes**

A marketing source code is a unique identifier that represents a campaign or marketing activity and is used to track marketing effectiveness. While a source code does not represent individual promotions or coupons, it does represent the campaign stage instance, audience segment or list, and treatment combination of a multistage campaign, and the instance, campaign content, and audience segment or list combination of a mini campaign.

To view a listing of source codes and associated campaign components, navigate to the Review Marketing Source Codes task in Functional Setup Manager. Note that a source code can be listed more than once in this page to support querying on the related promotions and coupons.

**Promotions, Coupons, and Marketing Responses**

You can provide a source code when creating, editing, or importing a marketing response. A response generated as a result of an e-mail campaign trackable URL or an e-mail bounce inherits the source code derived from the originating e-mail marketing campaign.

By providing a source code on the marketing response, associated promotions and coupons are also added as informative information on the response. You can search responses by promotion name and coupon code.
E-mail Campaign Content: Explained

To create the content of your e-mail campaign, you can add any combination of the elements that follow.

- Images
- Merge Fields
- Response Forms
- Standard, Dynamic, and Ad Hoc URLs
- Conditional Content
- Attachments

Images

Add graphic images to your e-mail. You can provide a link to an image already on a server, or choose a local file and upload it to the server. Specify the size and placement of the image.

Merge Fields

Personalize your e-mail message body by adding place holders from a list of merge fields, such as the recipient's first name. Merge fields are standard sets of attributes pertaining to an individual contact. You can insert them into your e-mail content. When you launch the campaign, merge fields are dynamically populated directly from the database.

The personalized text merge field, available only with sales campaigns, is the only exception. Rather than being populated from the database, you enter the content yourself. You can enter different text for every individual contact, to provide a personal touch to an otherwise generic collateral.

Response Forms

Insert response forms as active links in your e-mail content. When a contact clicks one of the links, a specific response is automatically generated. All such responses are gathered and monitored, to track contacts’ actions.

Available response forms for marketing campaigns are:

- Forward to Friend
- Request Call Back
- Request More Information
- Subscription Confirmation Request
• Unsubscribe Confirmation Request
• Do Not Contact
• Request Unsubscribe from Many
• Subscribe to List
• Unsubscribe from List

Note

Only the first three response forms are available in a sales campaign.

Standard, Dynamic, and Ad Hoc URLs

You can add any of the following types of URL:

• Standard URLs are predefined and commonly used across the enterprise.

• A dynamic URL is a type of standard URL where content is inserted when
  the e-mail is sent.

• An Ad Hoc URL can be created when you define your e-mail content.

If you enable tracking, every time a contact clicks a link in a campaign e-mail,
the click is recorded as a specific response, together with details pertaining to the
URL.

Conditional Content

Conditional content is inserted or omitted based on the results of a rule. There
are two elements that you can use to generate conditional content within an e-
mail:

• If-Then-Else cases are used to determine if a block of text or HTML is
  removed from an outgoing e-mail, by verifying if a Merge field value is
  defined. The merge field is defined if it is NOT an empty string or, for
  numeric merge fields, if it is not a value of 0 (zero). If the merge field is
  defined, the block of text or HTML remains in the outgoing e-mail. If the
  merge field is not defined, the block of text or HTML is removed from the
  outgoing e-mail.

• Named Blocks are used to decide whether or not to insert a block of text
  or HTML into an outgoing e-mail, by comparing one string to another. If
  the two strings are identical, the block is inserted.

Attachments

To send documents, such as product literature, as e-mail attachments with your
campaign, you can select a file from your own desktop; the file will be added to
the content repository for use during delivery.
Define Segmentation Manager

Marketing Segmentation: Highlights

Marketing integrates with Oracle Marketing Segmentation to provide customer segments, personalization merge fields, contact planning rules, and list formats.

Oracle Marketing Segmentation is fully integrated with Oracle Business Intelligence and uses the same metadata as the Business Intelligence reporting tools. Consequently, Marketing Segmentation has access to a library of available customer attributes, calculations, derived metrics, and data mining models.

Oracle Enterprise Manager incorporates the integration points between Oracle Marketing Segmentation and Universal Content Repository for the storage of lists generated for export.

Configuring segmentation for Marketing includes the management of the following tasks:

- Segmentation Manager Configuration Parameters
- Marketing List Formats
- Content Management Integration
- Contact Planning Rules
- Marketing Module Metadata

Segmentation Manager Configuration Parameters

- For information about setting up marketing module parameters, see the chapter on Installing and Administering Segmentation and List Generation in the Marketing Segmentation Guide.

Marketing List Formats

List Formats define the layout of files that you can use for a variety of purposes. While Marketing Segmentation supports five list format types, Oracle Fusion Marketing uses four formats. List Export Formats are used to define the customer data and other campaign-related information that is exported so that the campaign stage can be executed. For example, a list export format
may provide a list file containing customers and addresses to a direct mail fulfillment supplier for printing and mailing or to create a call list to distribute to employees in a sales organization. E-Mail Server Formats are used to export the members of an e-mail campaign to the Oracle E-Mail Marketing Server. The e-mail personalization format is the type of E-Mail Server format that provides the columns that can be used as merge fields to personalize the e-mail message sent to each recipient. Campaign Load Formats are used to load the individual members of a segment or a segment tree to the campaign. Data Load Formats are used to import leads resulting from a multistage campaign, lead generation stage.

- For information about marketing list formats, see the chapter on Designing Marketing List Formats in the Marketing Segmentation Guide.
- To provide default campaign load and data load formats for campaigns, navigate to the Functional Setup Manager and select the Manage Marketing Profile Options task to update the Campaign Member Load Format Default and Campaign Lead Generation Load Format Default profiles.

**Content Management Integration**

Configure deployment and connection parameters for Oracle Enterprise Content Management server repository using Oracle Enterprise Manager.

- For information about content management integration, see the Configuring for Connections to the Marketing Content Server section in the Configuring Connections to External Systems chapter of the Oracle Fusion Middleware System Administrator’s Guide for Oracle Business Intelligence Enterprise Edition.

**Contact Planning Rules**

Contact planning rules allow you to define contact frequency and volume constraints that can be applied globally or against a specific customer communication channel. Contact planning rules use segment inclusion criteria to define the conditions where the rules apply.

- For information about contact planning rules, see the Manage Marketing Contact Planning Rules section in the Using Marketing Segments and Segment Trees chapter in the Marketing Segmentation Guide.

**Marketing Module Metadata**

To support the segmentation process, the Oracle Business Intelligence Administration Tool provides a set of Marketing metadata, such as target levels, presentation catalogs, sampling factors, and schema mappings.

- For information about marketing module metadata, see the chapter on Configuring Marketing Module Metadata in the Marketing Segmentation Guide.
Define Transactional Business Intelligence Configuration

Define Transactional Business Intelligence Configuration: Highlights

Configure Oracle Transactional Business Intelligence for ad hoc reporting, review certain setup objects to be used in Transactional Business Intelligence, and manage the presentation catalog and currency type display.

Defining Transactional Business Intelligence Configuration

- Review details about the Transactional Business Intelligence tasks.
  
  See: Oracle Fusion Transactional Business Intelligence Administrator's Guide

Access to Person Data

Assigning Security Profiles to Job Roles for Oracle Fusion Transactional Business Intelligence Users: Explained

Users of Oracle Fusion Transactional Business Intelligence (Transactional Business Intelligence) need access to some person data for reporting purposes. To provide this access, you assign a predefined security profile to relevant job or abstract roles using the Oracle Fusion Human Capital Management (HCM) setup task Manage Data Role and Security Profiles. On completion of this task, Oracle Fusion Data Security is updated automatically for roles being used to access Transactional Business Intelligence.

Job or Abstract Roles and Related Security Profiles

The following table identifies, by Oracle Fusion product, the job and abstract roles that need access to person data and the predefined security profile that you assign to each role:

<table>
<thead>
<tr>
<th>Product</th>
<th>Job or Abstract Role</th>
<th>Security Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Fusion Assets</td>
<td>Asset Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Assets</td>
<td>Asset Accounting Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Oracle Fusion Enterprise Planning and Budgeting</td>
<td>Budget Analyst</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Enterprise Planning and Budgeting</td>
<td>Budget Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financial Consolidation Hub</td>
<td>Consolidation Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financial Consolidation Hub</td>
<td>Consolidation Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financials Common Module</td>
<td>Intercompany Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>Financial Analyst</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>General Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>General Accounting Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Incentive Compensation</td>
<td>Incentive Compensation Participant Manager</td>
<td>View Manager Hierarchy</td>
</tr>
<tr>
<td>Oracle Fusion Inventory Management</td>
<td>Warehouse Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Administrator</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Billing Specialist</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Purchasing</td>
<td>Buyer</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Sourcing</td>
<td>Category Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Sourcing</td>
<td>Sourcing Project Collaborator</td>
<td>View All Workers</td>
</tr>
</tbody>
</table>

For example, as part of their Transactional Business Intelligence setup:

- Oracle Fusion Assets implementors must assign the predefined security profile View All Workers to the Asset Accountant and Asset Accounting Manager job roles.

- Oracle Fusion Incentive Compensation implementors must assign the predefined security profile View Manager Hierarchy to the abstract role Incentive Compensation Participant Manager.

The security profiles that HCM roles need to access Transactional Business Intelligence are assigned during the setup of HCM data security: no additional setup is required for Transactional Business Intelligence purposes.

**Enabling an Oracle Fusion Transactional Business Intelligence User to Access Person Data: Worked Example**

This example shows how to assign a security profile to a job or abstract role to enable users with that role to access person data. This task is required for users of Oracle Fusion Transactional Business Intelligence (Transactional Business Intelligence) who do not also use Oracle Fusion Human Capital Management (HCM).
The following table summarizes key decisions for this scenario. When performing this task, use the job or abstract role for your product and the name of the relevant predefined person security profile in place of those shown here.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the name of the Transactional Business Intelligence job or abstract role?</td>
<td>Warehouse Manager</td>
</tr>
<tr>
<td>What is the name of the person security profile?</td>
<td>View All Workers</td>
</tr>
</tbody>
</table>

To perform these tasks, you must have the role IT Security Manager.

1. Launch the task Manage Data Role and Security Profiles.
2. Search for the job or abstract role.
3. Assign the relevant predefined security profile to the job or abstract role.

**Launching the Task Manage Data Role and Security Profiles**

1. Sign in to the Functional Setup Manager (FSM).
2. In the FSM, click the **All Tasks** tab.
3. In the Search region, complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Tasks</td>
</tr>
<tr>
<td>Name</td>
<td>Manage Data Role and Security Profiles</td>
</tr>
</tbody>
</table>

4. Click **Search**.
5. In the search results, click **Go to Task** for the Manage Data Role and Security Profiles task.

**Searching for the Job or Abstract Role**

1. On the Manage HCM Data Roles page, enter the job name Warehouse Manager in the **Role** field.
2. Click **Search**.
3. In the search results, highlight the entry for the Warehouse Manager job role.
4. Click **Assign**.

**Assigning the Security Profile to the Job Role**

1. In the **Person Security Profile** field on the Assign Data Role: Security Criteria page, select the security profile View All Workers.
2. Click **Review**.
3. On the Assign Data Role: Review page, click **Submit**.
Define Extensions for Marketing

Define Custom Enterprise Scheduler Jobs

Managing Job Definitions: Highlights

Oracle Enterprise Scheduler jobs are run in Oracle Fusion Applications to process data and, in some cases, to provide report output. A job definition contains the metadata that determines what the job does and what options are available to users who run the job. You can create and maintain job definitions for use in Oracle Fusion Applications.

Managing job definitions is fully described in the Oracle Fusion Applications Administrator's Guide. As you read content from that guide, note that the guide describes managing Oracle Enterprise Scheduler, including job definitions, from Oracle Enterprise Manager Fusion Applications Control. You can also access job definitions by starting in the Setup and Maintenance Overview page and searching for the Enterprise Scheduler job tasks for your applications.

Viewing and Creating Job Definitions

- You can access predefined and custom job definitions.
  See: Viewing Job Definitions

- You can create jobs based on Java, PL/SQL, or any other supported technology.
  See: Creating a Job Definition

- If you are using the Setup and Maintenance work area, then the Enable submission from Enterprise Manager check box is available for the job definition.
  - If you do not select this check box, then the job cannot be run from Enterprise Manager.
  - If you select this check box, then you can define parameters for this job definition only in Enterprise Manager. Save the rest of your work on the job definition, and then go to Enterprise Manager if you need to define parameters.

Managing List of Values Sources: Highlights

A list of values source for Oracle Enterprise Scheduler job definitions determines where a list of values comes from and what the specific values are. These lists of values are used in parameters and user properties of job definitions. For example, you can use a source of country values for a Country job parameter.

Managing list of values sources is fully described in the Oracle Fusion Applications Administrator's Guide. As you read content from that guide,
note that the guide describes managing Oracle Enterprise Scheduler, including list of values sources, from Oracle Enterprise Manager Fusion Applications Control. You can also access list of values sources by starting in the Setup and Maintenance Overview page and searching for Enterprise Scheduler job tasks.

**Registering and Searching for List of Values Sources**

- Create list of values sources to register them for use in job definitions.
  See: Registering Sources for Lists of Values

- Search for list of values sources to edit or delete, or to make sure a particular source does not already exist before you create it.
  See: Searching for List of Value Sources

**Customization and Sandboxes**

**Customizing Pages Using Oracle Composer: Highlights**

You can customize dashboards and some work areas, where available, for all or some users based on a selected customization layer, for example only for users in a specific country or with a specific job role. When you select to customize a page from the Administration menu in the global area, you invoke Oracle Composer, which enables the customization. The Administration menu is only available if you have appropriate roles.

From the same menu, you can also access the Customization Manager, which displays a list of components in the current page and details about the layers in which each component is customized. You can also access sandboxes from the Administration menu, to make customizations to a runtime use session before deploying your changes to the mainline.

Customizing pages using Oracle Composer, the Customization Manager, and sandboxes are described in the Oracle Fusion Applications Extensibility Guide.

**Editing Pages**

- Customizations that you can make to existing pages include adding content and changing layout.
  See: Editing a Page in Oracle Composer

- You can also update display and other options for specific components within the page.
  See: Editing Component Properties in Oracle Composer

**Customization Manager**

- Use Customization Manager to analyze and diagnose customization metadata, and to perform customization related tasks that cannot be done in the user interface, for example to revert customizations to a previous version. You can also do direct customization by manipulating the metadata and uploading it back.
  See: Using Customization Manager to Manage Runtime Customizations

**Sandboxes**

- Create or select an appropriate sandbox, and set it as active to capture your customizations using Oracle Composer. When you are ready, you publish the sandbox to make your changes available to users.
  See: Using the Sandbox Manager
What’s the difference between Oracle Composer and Oracle Fusion CRM Application Composer?

Oracle Composer lets you make user interface changes at run time, such as hide or show fields and regions, across all Oracle Fusion applications. You can also customize a page by adding new content from the Resource Catalog.

The Oracle Fusion CRM Application Composer also lets you make user interface changes at run time. However, the types of user interface changes that you can make using the CRM Application Composer are quite different. Specifically, your primary focus when using the CRM Application Composer is to make actual object model changes. For example, you can create a new business object and related fields, and then create new application pages where that object and its fields are exposed to users. The ability to make these types of object model extensions is available only in Oracle Fusion CRM applications. Also, using the CRM Application Composer, you cannot access the Resource Catalog to add new content to a page.

This table describes some of the primary differences between Oracle Composer and the CRM Application Composer:

<table>
<thead>
<tr>
<th>Customization Task</th>
<th>Available in Oracle Composer?</th>
<th>Available in Oracle Fusion CRM Application Composer?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make object model extensions and expose your customizations by creating or modifying work area pages</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Reorder subtabs</td>
<td>Yes, on a page-by-page basis, if the page author enabled this feature.</td>
<td>Yes</td>
</tr>
<tr>
<td>Reorder tree nodes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Customize dashboard pages</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Add content from the Resource Catalog</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Select the MDS layer where you want to author customizations, such as at the site layer or job role layer</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>View results of customizations immediately</td>
<td>Yes, in the Oracle Composer design interface</td>
<td>Yes, in the CRM application that you are customizing</td>
</tr>
<tr>
<td>Apply specific styling (such as font, color, and size) to user interface components</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Customizing Oracle Fusion CRM Applications Using Oracle Fusion CRM Application Composer: Explained

The Oracle Fusion CRM Application Composer provides a series of task flows which let you customize and extend an Oracle Fusion CRM application according to the needs of your users. For example, you can create new fields for an existing standard object, and expose those new fields on the object’s work area. Or, create a brand new custom object and related fields, then create a work area where that object and its fields are exposed to users. The task flows available to you are dependent upon the CRM application that you are
customizing. This topic provides an overview of which CRM Application Composer task flows are available for use in each CRM application. This topic addresses extensibility for these CRM applications:

- Oracle Fusion Common CRM
- Oracle Fusion Customer Center
- Oracle Fusion Marketing
- Oracle Fusion Sales
- Oracle Fusion Sales Catalog

You can also refer to the product-specific implementation guides to learn more about how a particular application works with the Application Composer.

### Oracle Fusion Common CRM

The creation of custom objects is not supported for the Oracle Fusion Common CRM application.

For Oracle Fusion Common CRM standard objects, you can do the following in the Application Composer:

<table>
<thead>
<tr>
<th>Customization</th>
<th>Trading Community Org Contact</th>
<th>Trading Community Resource Profile</th>
<th>Trading Community Organization Profile</th>
<th>Trading Community Organization Address</th>
<th>Activity Task</th>
<th>Interaction</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create and expose custom fields on existing pages that are available for extensibility</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Edit display label and help text of standard fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom currency fields</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Index custom fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add custom buttons (Groovy script or URL) to selected pages</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add links (URL) to selected pages</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature Description</td>
<td>Yes</td>
<td>No</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------------------------------------------</td>
<td>-----</td>
<td>-----</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create and expose custom child objects on an object's details page</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Not applicable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom field-level and object-level validation logic (Groovy scripts)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom logic at various object trigger points (Groovy scripts)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom saved searches at the site level</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provide Mobile page support</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom relationships</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manage object workflows</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Web services</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Import data using file-based import</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Export data using bulk export</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom subject areas</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Oracle Fusion Customer Center

You can create custom objects for the Oracle Fusion Customer Center application.

For Oracle Fusion Customer Center’s standard object, you can do the following in the Application Composer:

<table>
<thead>
<tr>
<th>Customization Task Flow</th>
<th>Sales Account Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create and expose custom fields on existing pages that are available for extensibility</td>
<td>Yes</td>
</tr>
<tr>
<td>Edit display label and help text of standard fields</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom currency fields</td>
<td>No</td>
</tr>
<tr>
<td>Index custom fields</td>
<td>Yes</td>
</tr>
<tr>
<td>Add custom buttons (Groovy script or URL) to selected pages</td>
<td>No</td>
</tr>
<tr>
<td>Add links (URL) to selected pages</td>
<td>No</td>
</tr>
<tr>
<td>Create and expose custom child objects on an object’s details page</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom field-level and object-level validation logic (Groovy scripts)</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom logic at various object trigger points (Groovy scripts)</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom saved searches at the site level</td>
<td>No</td>
</tr>
<tr>
<td>Provide Mobile page support</td>
<td>No</td>
</tr>
<tr>
<td>Create custom relationships</td>
<td>Yes</td>
</tr>
<tr>
<td>Manage object workflows</td>
<td>Yes</td>
</tr>
<tr>
<td>Web services</td>
<td>Yes</td>
</tr>
<tr>
<td>Import data using file-based import</td>
<td>Yes</td>
</tr>
<tr>
<td>Export data using bulk export</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom subject areas</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Oracle Fusion Marketing

You can create custom objects for the Oracle Fusion Marketing application.

For Oracle Fusion Marketing standard objects, you can do the following in the Application Composer:
<table>
<thead>
<tr>
<th>Create and expose custom fields on existing pages that are available for extensibility</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit display label and help text of standard fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom currency fields</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Index custom fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add custom buttons (Groovy script or URL) to selected pages</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add links (URL) to selected pages</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create and expose custom child objects on an object’s details page</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Create custom field-level and object-level validation logic (Groovy scripts)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create custom logic at various object trigger points (Groovy scripts)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom saved search at the site level</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Provide Mobile page support</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Create custom relationship</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Manage object workflows</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Web services</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Import data using file-based import</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Export data using bulk export</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Create custom subject areas</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Oracle Fusion Sales

You can create custom objects for the Oracle Fusion Sales application. For Oracle Fusion Sales standard objects, you can do the following in the Application Composer:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Create and expose custom fields on existing pages that are available for extensibility</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Edit display label and help text of standard fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom currency fields</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Index custom fields</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add custom buttons (Groovy script or URL) to selected pages</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add links (URL) to selected pages</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create and expose custom child objects on an object’s details page</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Create custom field-level and object-level validation logic (Groovy scripts)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom logic at various object trigger points (Groovy scripts)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom saved searches at the site level</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Provide Mobile page support</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom relationships</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Feature</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>Manage object workflows</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Web services</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Import data using file-based import</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Export data using bulk export</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom subject areas</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Oracle Fusion Sales Catalog**

The creation of custom objects is not supported for the Oracle Fusion Sales Catalog application.

For Oracle Fusion Sales Catalog’s standard object, you can do the following in the Application Composer:

<table>
<thead>
<tr>
<th>Customization Task Flow</th>
<th>Product Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create and expose custom fields on existing pages that are available for extensibility</td>
<td>Yes</td>
</tr>
<tr>
<td>Edit display label and help text of standard fields</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom currency fields</td>
<td>No</td>
</tr>
<tr>
<td>Index custom fields</td>
<td>Yes</td>
</tr>
<tr>
<td>Add custom buttons (Groovy script or URL) to selected pages</td>
<td>No</td>
</tr>
<tr>
<td>Add links (URL) to selected pages</td>
<td>No</td>
</tr>
<tr>
<td>Create and expose custom child objects on an object’s details page</td>
<td>No</td>
</tr>
<tr>
<td>Create custom field-level and object-level validation logic (Groovy scripts)</td>
<td>No</td>
</tr>
<tr>
<td>Create custom logic at various object trigger points (Groovy scripts)</td>
<td>No</td>
</tr>
<tr>
<td>Create custom saved searches at the site level</td>
<td>No</td>
</tr>
<tr>
<td>Provide Mobile page support</td>
<td>No</td>
</tr>
<tr>
<td>Create custom relationships</td>
<td>No</td>
</tr>
<tr>
<td>Manage object workflows</td>
<td>No</td>
</tr>
<tr>
<td>Web services</td>
<td>No</td>
</tr>
<tr>
<td>Import data using file-based import</td>
<td>Yes</td>
</tr>
<tr>
<td>Export data using bulk export</td>
<td>Yes</td>
</tr>
<tr>
<td>Create custom subject areas</td>
<td>No</td>
</tr>
</tbody>
</table>

**Customizing Oracle Fusion Customer Center Pages Using Oracle Composer: Explained**

In Oracle Fusion Customer Center, you can customize the Sales Account region on the customer profile page using Oracle Composer. To access Oracle Composer, navigate to the customer profile page and select to customize the
page from the Administration menu in the global area. You can also access Oracle Composer by selecting Customization Manager from the Administration menu.

Available Customization Options
When you select to customize a page from the Administration menu in the global area, you launch Oracle Composer.
The customizations that you make to the customer profile are applied based on your layer selection:

- **Site**
  Your customizations are visible to all users.

- **External or Internal**
  Depending on your selection, your customizations are visible to either external or internal users.
  External users could be your partners or anonymous users. Internal users could be your employees.

- **Job Role**
  Your customizations are visible to users who have the selected job role.
  If a user has more than one job role, then the sequence in which customizations are applied is alphabetically ordered by job role name.

Oracle Composer provides two views for working with page content: Design View and Source View. Design View provides a WYSIWYG rendering of the page and its content, where controls are directly selectable on each component. Source View provides a combined WYSIWYG and hierarchical rendering of page components, where controls are available on the header of the hierarchical list. Source View provides access to page layout components that are otherwise not exposed on the page, and therefore not available in Design View.

This table lists the types of customizations available for the customer profile page. You can perform most of the basic customizations in either Design View or Source View. Some customizations, however, must be completed only in Source View.

<table>
<thead>
<tr>
<th>Oracle Fusion Customer Center Page</th>
<th>Customization Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer profile</td>
<td>Show and hide components on a page:</td>
</tr>
<tr>
<td></td>
<td>• Show and hide fields within the Sales Account region by selecting the <strong>Show Component</strong> check box on the <strong>Display Options</strong> tab in the Component Properties dialog for a component.</td>
</tr>
<tr>
<td></td>
<td>• Show and hide the Sales Account region itself by selecting the <strong>Show Component</strong> check box on the <strong>Display Options</strong> tab in the Component Properties dialog for the container.</td>
</tr>
<tr>
<td></td>
<td><strong>Tip</strong> You might have to scroll down to see the <strong>Show Component</strong> check box.</td>
</tr>
</tbody>
</table>
Customer profile

Move components on a page:

- Move components within the Sales Account region (only) by cutting and pasting them in Source View, or by rearranging them on the Child Components tab in the Component Properties dialog for the container.

**Note**
The components available for updating include any custom fields that were added to the Sales Account object using the CRM Application Composer.

Customer profile

Make an updateable field read only by selecting the Read Only check box on the Display Options tab in the Component Properties dialog.

Customer profile

Make a field required by selecting the Show Required check box on the Display Options tab in the Component Properties dialog.

**Customization Privileges Required**

The customizations that you can make using the Oracle Composer Source View are available to you only if your assigned job role includes the **Page Composer Source View Access Duty** duty role. Contact your security administrator for details.

**Note**
Your assigned job role must also include the **Administration Link View Duty** duty role to access Oracle Composer at all, even if only the Design View. This duty role exposes the Administration menu in the global area.

**Sandboxes: Highlights**

Use a sandbox to commit customizations to a runtime use session for validation before deploying changes to the mainline. Administrators create and manage sandboxes. An active sandbox isolates changes from the mainline and other users.

Sandboxes can contain the following types of customization changes.

- Metadata, such as non-flexfield UI page customizations
- Data security
- Generated flexfields business components

Metadata changes are captured in a metadata sandbox. Data security changes are additionally captured in a data security enabled sandbox. Changes to a flexfield are captured in a flexfield that is deployed as a single flexfield sandbox. Once
you are ready to make sandbox changes available in the mainline, you either publish the metadata or data security sandbox, or deploy the flexfield. Only metadata and data security sandboxes can be downloaded as a sandbox file for import to another Oracle Fusion Applications instance.

The following table lists the differences among the types of sandboxes.

<table>
<thead>
<tr>
<th>Type of Changes</th>
<th>Type of Sandbox</th>
<th>Method for Making Changes Available in Mainline</th>
<th>Downloadable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>Sandbox</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Data security</td>
<td>Sandbox enabled for data security changes</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Flexfield</td>
<td>Flexfield deployed as a flexfield-enabled sandbox</td>
<td>Deploy flexfield</td>
<td>No</td>
</tr>
</tbody>
</table>

Only one sandbox can be active at a time. Changes made while a sandbox is active are captured in that sandbox.

For more information on using the Sandbox Manager, and customizing and securing pages, business objects, data, and custom objects in a sandbox, see the Oracle Fusion Applications Extensibility Guide.

**Managing a Page Customization Sandbox**

You can make metadata (MDS) type changes in a sandbox, including menu customizations, changes to the personalization menu, implicit ADF customizations, or changes made with Oracle Composer or CRM Application Composer.

- If you are entitled to do so, manage sandboxes in the Sandbox Manager.

  See: Using the Sandbox Manager

  - Implement customizations on an existing page to change the metadata of a sandbox before deploying the changes to the mainline.

    See: Customizing Existing Pages

  - Using CRM Application Composer, customize business objects in a sandbox before deploying the changes to the mainline.

    See: Customizing Objects

**Managing a Data Security Sandbox**

You can create a sandbox for data security testing, or designate an existing sandbox to become enabled for data security testing.

- If you are entitled to do so, manage data security-enabled sandboxes in the Sandbox Manager.

  See: Using the Sandbox Manager
• If you customize business objects in CRM Application Composer, you may need to define data security policies to make them accessible to users.

See: Defining Security Policies for Business Objects

• If you create new business objects, you need to secure them.

See: Customizing Security for Custom Business Objects

Managing a Flexfield Sandbox

You create a flexfield-enabled sandbox by deploying one flexfield to a sandbox using the Manage Flexfield task flow. The flexfield sandbox gets its name from the flexfield you deploy. You cannot test two flexfields in the same sandbox. Once you deploy a flexfield as a sandbox, you must sign out and back in to view how the sandbox runtime reflects the flexfield changes, such as new segments. You can redeploy the same flexfield to the same sandbox repeatedly as you make incremental changes to the flexfield setup.

• Since a flexfield sandbox cannot be published, any page customizations or data security in the flexfield sandbox cannot reach the mainline when the flexfield is deployed to the mainline. If you have entitlement to do so, see Deploying a Flexfield to a Sandbox: Points to Consider.

• If you are entitled to do so, manage flexfield-enabled sandboxes in the Sandbox Manager.

See: Using the Sandbox Manager
Importing and Exporting Setup Data

Configuration Packages: Explained

Almost all Oracle Fusion application implementations require moving functional setup data from one instance into another at various points in the lifecycle of the applications. For example, one of the typical cases in any enterprise application implementation is to first implement in a development or test application instance and then deploy to a production application instance after thorough testing. You can move functional setup configurations of applications from one application instance into another by exporting and importing Configuration packages from the Manage Configuration Packages page.

A Configuration Package contains the setup import and export definition. The setup import and export definition is the list of setup tasks and their associated business objects that identifies the setup data for export as well as the data itself. When you create a configuration package only the setup export and import definition exists. Once you export the configuration package appropriate setup data is added to the configuration package using the definition. Once a configuration package is exported, the setup export and import definition is locked and cannot be changed.

You generate the setup export and import definition by selecting an implementation project and creating a configuration package. The tasks and their associated business objects in the selected implementation project define the setup export and import definition for the configuration package. In addition, the sequence of the tasks in the implementation project determine the export and import sequence.

Exporting and Importing Setup Data: Explained

A configuration package is required to export setup data. You can export a configuration package once you create it, or at any time in the future. During export, appropriate setup data will be identified based on the setup export definition and added to the configuration package. The setup data in the configuration package is a snapshot of the data in the source application instance at the time of export. After the export completes, you can download the configuration package as a zipped archive of multiple XML files, move it to the target application instance, and upload and import it.

Export

You can export a configuration package multiple times by creating multiple versions. While the export definition remains the same in each version, the setup data can be different if you modified the data in the time period between the different runs of the export process. Since each version of the configuration package has a snapshot of the data in the source instance, you can compare and
analyze various versions of the configuration package to see how the setup data changed.

Import
In the target application instance, the setup import process will insert all new data from the source configuration package that does not already exist and update any existing data with changes from the source. Setup data that exists in the target instance but not in source will remain unchanged.

Export and Import Reports
You can review the results of the export and import processes using reports. The results appear ordered by business objects and include information on any errors encountered during the export or import process. If a setup export or import process paused due to errors encountered or for a manual task to be performed outside of the application, then you can resume the paused process. These reports show what setup data was exported or imported and by which specific process. You can change the reports to validate the setup data as well as to compare or analyze it. A report is generated for each business object. These reports show the same information as the export and import results seen directly in the application.

Process status details are available as text files showing the status of an export or import process including the errors encountered during the process.

Moving Common Reference Objects

Moving Common Reference Objects: Overview
The common reference objects in Oracle Fusion Middleware Extensions for Applications are used by several setup tasks in the Setup and Maintenance work area. The common reference objects become a part of the configuration package that is created for an implementation project. While moving the application content, for example, from the test phase to the production phase of an implementation, you must pay special attention to the nuances of these common reference objects.

Parameters
The common reference objects are represented as business objects. A single object can be referenced in multiple setup tasks with different parameters. In the configuration package that is created for the implementation project, parameters passed to a setup task are also passed to the business objects being moved. As a result, the scope of the setup tasks is maintained intact during the movement.

Dependencies
Common reference objects may have internal references or dependencies among other common reference objects. Therefore, it is necessary that all the dependencies are noted before the movement of objects so that there are no broken references among the objects.

Business Objects for Moving Common Reference Objects: Points to Consider
Common reference objects in Oracle Fusion Functional Setup Manager are represented by business objects. These business objects are the agents that contain the application content and carry them across whenever the application setup is moved from one environment to another, for example, test environment to production environment.

Choice of Parameters
The following table lists the business objects, the corresponding movement details, and the effect of the setup task parameter on the scope of the movement.
Note

Only the translation in the current user language is moved.

<table>
<thead>
<tr>
<th>Business Object Name</th>
<th>Moved Functional Item</th>
<th>Effect on the Scope of Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Menu Customization</td>
<td>Customizations to the navigator menu</td>
<td>Movement of navigator menu customizations requires manual effort. For details regarding the manual command, refer to the Oracle Fusion Applications Administrators Guide.</td>
</tr>
<tr>
<td>Application Message</td>
<td>Messages and associated tokens</td>
<td>No parameters: all messages are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>messageName/applicationId: only the specified message is moved.</td>
</tr>
<tr>
<td>Application Taxonomy</td>
<td>Application taxonomy modules and components</td>
<td>No parameters: all taxonomy modules and components are moved.</td>
</tr>
<tr>
<td>Application Attachment Entity</td>
<td>Attachment entities</td>
<td>No parameters: all attachment entities are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only attachment entities belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td>Application Attachment Category</td>
<td>Attachment categories and category-to-entity mappings</td>
<td>No parameters: all attachment categories and category-to-entity mappings are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only attachment categories belonging to the specified module and its descendant modules in the taxonomy hierarchy along with the respective category-to-entity mappings are moved.</td>
</tr>
<tr>
<td>Application Document Sequence Category</td>
<td>Document sequence categories</td>
<td>No parameters: all categories are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>code/applicationId: only the specified document sequence category code is moved.</td>
</tr>
</tbody>
</table>
| Application Document Sequence | Document sequences and their assignments | No parameters: all sequences are moved.  

**moduleType/moduleKey**: only document sequences belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved  
**name**: only the specified document sequence is moved. |
| Application Descriptive Flexfield | Descriptive flexfield registration data and setup data | No parameters: all descriptive flexfields are moved.  

**moduleType/moduleKey**: only descriptive flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
**descriptiveFlexfieldCode/applicationId**: only the specified descriptive flexfield is moved. |
| **Note** |  
Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment. |
| Application Extensible Flexfield | Extensible flexfield registration data and setup data, including categories | No parameters: all extensible flexfields are moved  

**moduleType/moduleKey**: only extensible flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
**extensibleFlexfieldCode/applicationId**: only the specified extensible flexfield is moved. |
| **Note** |  
Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment. |
<table>
<thead>
<tr>
<th>Application</th>
<th>Flexfield/Value Set</th>
<th>Setup Data</th>
<th>Parameters</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Key Flexfield</td>
<td>Key flexfield registration data and setup data</td>
<td>No parameters: all key flexfields are moved.</td>
<td>moduleType/moduleKey: only key flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. keyFlexfieldCode/applicationId: only the specified key flexfield is moved.</td>
<td>Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.</td>
</tr>
<tr>
<td>Application Flexfield Value Set</td>
<td>Value set setup data</td>
<td>No parameters: all value sets are moved.</td>
<td>moduleType/moduleKey: only value sets belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. valueSetCode: only the specified value set is moved.</td>
<td>Importing a value set’s metadata can change the deployment status of flexfields that use the value set, and therefore the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.</td>
</tr>
<tr>
<td>Application Reference Currency</td>
<td>Currency data</td>
<td>No parameters: all currencies are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference ISO Language</td>
<td>ISO language data</td>
<td>No parameters: all ISO languages are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference Industry</td>
<td>Industry data including industries in territories data</td>
<td>No parameters: all industries are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference Language</td>
<td>Language data</td>
<td>No parameters: all languages are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference Natural Language</td>
<td>Natural language data</td>
<td>No parameters: all natural languages are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference Territory</td>
<td>Territory data</td>
<td>No parameters: all territories are moved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>----------------</td>
<td>----------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Reference Time zone</td>
<td>Time zone data</td>
<td>No parameters: all time zones are moved.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Application Standard Lookup     | Standard lookup types and their lookup codes | No parameters: all standard lookups are moved.  
moduleType/moduleKey: only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
lookupType: only the specified common lookup is moved. |
| Application Common Lookup       | Common lookup types and their lookup codes | No parameters: all common lookups are moved.  
moduleType/moduleKey - only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
lookupType: only the specified common lookup is moved. |
| Application Set-Enabled Lookup  | Set-enabled lookup types and their lookup codes | No parameters: all set-enabled lookups are moved.  
moduleType/moduleKey: only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
lookupType: only the specified set-enabled lookup is moved. |
| Application Profile Category    | Profile categories | No parameters: all profile categories are moved.  
moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
name/applicationId: only the specified category is moved. |
### Importing and Exporting Setup Data

#### Application Profile Option
- **Profile options and their values**
  - **No parameters**: all profile options and their values are moved.
  - **moduleType/moduleKey**: only profile options and their values belonging to the specified module are moved.
  - **profileOptionName**: only the specified profile option and its values are moved.

#### Application Profile Value
- **Profile options and their values**
  - **No parameters**: all profiles and their values are moved.
  - **moduleType/moduleKey**: only profiles and their values belonging to the specified module are moved.
  - **categoryName/categoryApplicationId**: only profiles and their values belonging to the specified category are moved.
  - **profileOptionName**: only the specified profile and its values are moved.

#### Application Reference Data Set
- **Reference data sets**
  - **No parameters**: all sets are moved.

#### Application Reference Data Set Assignment
- **Reference data set assignments**
  - **determinantType**: only assignments for the specified determinant type are moved.
  - **determinantType/referenceGroupName**: only assignments for the specified determinant type and reference group are moved.

#### Application Tree Structure
- **Tree structures and any labels assigned to the tree structure**
  - **No parameters**: all tree structures (and their labels) are moved.
  - **moduleType/moduleKey**: only tree structures (and their labels) belonging to the specified module are moved.
  - **treeStructureCode**: only the specified tree structure (with its labels) is moved.
| Application Tree       | Tree codes and versions | No parameters: all trees are moved.  
|                       |                        | moduleType/moduleKey: only trees belonging to the specified module are moved.  
|                       |                        | treeStructureCode: only trees belonging to the specified tree structure are moved.  
|                       |                        | TreeStructureCode/TreeCode: only trees belonging to the specified tree structure and tree code are moved.  
| Application Tree Label | Tree structures and any labels assigned to the tree structure | No parameters: all tree structures (and their labels) are moved.  
|                       |                        | moduleType/moduleKey: only tree structures (and their labels) belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
|                       |                        | treeStructureCode: only the specified tree structure (with its labels) is moved.  
| Application Data Security Policy | Database resources, actions, conditions, and data security policies | No parameters: all database resources/actions/conditions/policies are moved.  
|                       |                        | moduleType/moduleKey: only database resources/actions/conditions/policies belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
|                       |                        | objName: only the specified database resource along with its actions/conditions/policies is moved.  
| Note                  |                        | • If the policies being moved contain reference to newly created roles, move the roles before moving the policies.  
|                       |                        | • If the source and target systems use different LDAPs, manually perform the GUID reconciliation after moving the data security policies.  
| Application Activity Stream Configuration | Activity stream options | No parameters: all activity stream options are moved.  

Moving Related Common Reference Objects: Points to Consider

Certain common reference objects may use other common reference objects creating dependencies among the objects. During the movement of common reference objects, these dependencies or references need to be taken care of.

Dependencies

The dependencies among the common reference objects may be caused by any of the following conditions.

- Flexfield segments use value sets
- Value sets may make use of standard, common, or set-enabled lookups
- Key flexfields may have an associated tree structure and key flexfield segments may have an associated tree code
- Tree codes and versions may be defined over values of a value set
- Data security policies may be defined for value sets that have been enabled for data security

You may choose to move one, some, or all of the business objects by including the ones you want to move in your configuration package. For example, you may choose to move only value sets and not lookups, or you may choose to move both value sets and their lookups as part of the same package. Whatever be the combination, it is recommended that during the movement of objects, you follow an order that maintains the dependencies among the objects.

While moving the business objects, adhere to the guidelines and exactly follow the order as listed below.

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.

Note
In scenarios where there may be dependencies on other objects, you must move the dependencies before moving the referencing object. For example, if data security policies being moved have dependencies on newly created security roles, you must move the security roles before moving the security policies.

### Using Seed Data Framework to Move Common Reference Objects: Points to Consider

To move the common reference objects, you can use the Seed Data Framework (SDF). You can also use the command line interface of SDF to move the object setup data. For more information about seed data loaders including common reference object loaders, see Oracle Fusion Applications Developer’s Guide.

**Movement Dependencies**

The seed data interface moves only the setup metadata. For example, if you use SDF to import flexfield metadata, the flexfield setup metadata is imported into your database. However, you must invoke the flexfield deployment process separately after seed data import to regenerate the runtime flexfield artifacts in the target environment. Similarly, if you use SDF to import data security metadata, you must first move any new referenced roles and then manually run the GUID reconciliation where required.

To ensure that the reference data is not lost during the movement, certain guidelines are prescribed. It is recommended that you perform the movement of object data exactly in the order given below.

**Note**

Only the translation in the current user language is moved.

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.
8. Move created reference data sets before moving any objects that reference them, such as reference data set assignments and set-enabled lookups.
9. Move created document sequence categories before moving any objects that reference them, such as document sequences.

10. Move created tree labels before moving any objects that reference them, such as trees.

11. Move created data security objects and policies before moving any objects that reference them, such as value sets.

12. Move created value sets before moving any objects that reference them, such as flexfields.

13. Move created trees before moving any objects that reference them, such as key flexfields.
abstract role
A description of a person’s function in the enterprise that is unrelated to the person’s job (position), such as employee, contingent worker, or line manager. A type of enterprise role.

accounting flexfield
The chart of accounts that determines the structure, such as the number and order of individual segments, as well as the corresponding values per segment.

action
The kind of access named in a security policy, such as view or edit.

ADF
Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

application identity
Predefined application level user with elevated privileges. An application identity authorizes jobs and transactions for which other users are not authorized, such as a payroll run authorized to access a taxpayer ID while the user who initiated the job is not authorized to access such personally identifiable information.

application module
An application module is the transactional component that UI clients use to work with application data. It defines an updatable data model and top-level procedures and functions (called service methods) for a logical unit of work related to an end-user task.

application module class
Includes service methods or extended code required by the application module configuration.

application role
A role specific to applications and stored in the policy store.

Applications Core
Abbreviation for Oracle Fusion Middleware Extensions for Applications. The technical product code is FND.

assignment
A set of information, including job, position, pay, compensation, managers, working hours, and work location, that defines a worker’s or nonworker’s role in a legal employer.
balancing segment
A chart of accounts segment used to automatically balance all journal entries for each value of this segment.

beneficiary
A person or organization designated to receive benefits from a compensation plan on the death of the plan participant.

BPEL
Business Process Execution Language; a standard language for defining how to send XML messages to remote services, manipulate XML data structures, receive XML messages asynchronously from remote services, manage events and exceptions, define parallel sequences of execution, and undo parts of processes when exceptions occur.

business function
A business process, or an activity that can be performed by people working within a business unit and describes how a business unit is used.

business object
A resource in an enterprise database, such as an invoice or purchase order.

business unit
A unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy.

calendar event
A period that signifies an event, such as a public holiday or a training course, that impacts worker availability.

chart of accounts
The account structure your organization uses to record transactions and maintain account balances.

competency
Any measurable behavior required by an organization, job, or position that a person may demonstrate in the work context. A competency can be a piece of knowledge, a skill, an attitude, or an attribute.

condition
An XML filter or SQL predicate WHERE clause in a data security policy that specifies what portions of a database resource are secured.
**confirm subscribe request**
A marketing response form that can be inserted into an e-mail. When e-mail recipients click the link, a second e-mail is sent, requesting them to confirm their intent to subscribe to a list.

**confirm unsubscribe request**
A marketing response form that can be inserted into an e-mail. When e-mail recipients click the link, a second e-mail is sent, requesting them to confirm their intent to unsubscribe from a list.

**content item**
An individual quality, skill, or qualification within a content type that you track in profiles.

**content library**
A repository of the content types and individual content items that can be associated with person profiles and profiles for workforce structures such as jobs and positions.

**content type**
An attribute such as a skill, quality, or qualification that is added to a profile.

**context**
A grouping of flexfield segments to store related information.

**context segment**
The flexfield segment used to store the context value. Each context value can have a different set of context-sensitive segments.

**context-sensitive segment**
A flexfield segment that may or may not appear depending upon a context such as other information that has been captured. Context-sensitive segments are custom attributes that apply to certain entity rows based on the value of the context segment.

**contingent worker**
A self-employed or agency-supplied worker. Contingent worker work relationships with legal employers are typically of a specified duration. Any person who has a contingent worker work relationship with a legal employer is a contingent worker.

**corporate rate type**
Rate you define to standardize rates used in conversion of one currency to another over a period of time. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.
cost center
A unit of activity or group of employees used to assign costs for accounting purposes.

cost organization
A grouping of inventory organizations that indicates legal and financial ownership of inventory, and which establishes common costing and accounting policies.

country holding company
A legal entity that acts on behalf of several divisions within an enterprise, and is the legal employer in a country.

data dimension
A stripe of data accessed by a data role, such as the data controlled by a business unit.

data instance set
The set of human capital management (HCM) data, such as one or more persons, organizations, or payrolls, identified by an HCM security profile.

data role
A role for a defined set of data describing the job a user does within that defined set of data. A data role inherits job or abstract roles and grants entitlement to access data within a specific dimension of data based on data security policies. A type of enterprise role.

data role template
A template used to generate data roles by specifying which base roles to combine with which dimension values for a set of data security policies.

data security
The control of access to data. Data security controls what action a user can taken against which data.

data security policy
A grant of entitlement to a role on an object or attribute group for a given condition.

database resource
An applications data object at the instance, instance set, or global level, which is secured by data security policies.
**deal size**
Total monetary amount the customer is expected to spend.

**department**
A division of a business enterprise dealing with a particular area of activity.

**descriptive flexfield**
Customizable expansion space, such as fields used to capture additional descriptive information or attributes about an entity, such as customer cases. Information collection and storage may be configured to vary based on conditions or context.

**descriptive flexfield**
An extendable field that captures additional information.

**determinant**
A value that determines which reference data set will be used in a specific business context.

**determinant type**
Designates the field within transactional columns that controls how data is shared across organizations such as business unit, asset book, cost organization or project unit. The type determines the reference data sets that would be used in a transaction.

**determinant type**
An additional and optional field within transactional columns (besides category and application) that is used to assign document sequences. The available determinant types are Business Unit, Ledger, Legal Entity, and Tax Registration.

**determinant value**
A value specific to the determinant type dimension of a document sequence. The determinant value is relevant in a document sequence assignment only if the document sequence has a determinant type. If Ledger is the determinant type for a document sequence, the determinant value is the specific ledger number whose documents are numbered by the document sequence.

**dimension**
A data category used to define territory boundaries, such as geography. Dimensions contain related members usually organized in hierarchies. For example, a geography dimension often includes members such as countries and cities that belong to countries. Defined dimensions determine how to assign objects such as sales accounts, leads, and opportunities.

**dimension member**
Individual components of a dimension.
division
A business-oriented subdivision within an enterprise. Each division is organized to deliver products and services or address different markets.

document sequence
A unique number that is automatically or manually assigned to a created and saved document.

document type
A categorization of person documents that provides a set of options to control what document information to retain, who can access the documents, whether the documents require approval, and whether the documents are subject to expiry. A document type exists for a combination of document category and subcategory.

duty role
A group of function and data privileges representing one duty of a job. Duty roles are specific to applications, stored in the policy store, and shared within an Oracle Fusion Applications instance.

educational establishment
A school, college, university, or other learning institution.

employment terms
A set of information about a nonworker's or employee's job, position, pay, compensation, working hours, and work location that all assignments associated with the employment terms inherit.

enterprise
An organization with one or more legal entities under common control.

enterprise role
Abstract, job, and data roles are shared across the enterprise. An enterprise role is an LDAP group. An enterprise role is propagated and synchronized across Oracle Fusion Middleware, where it is considered to be an external role or role not specifically defined within applications.

entitlement
Grants of access to functions and data. Oracle Fusion Middleware term for privilege.

ESS
Acronym for Enterprise Storage Server. An application that optimizes data storage.
extensible flexfield

Customizable expansion space, as with descriptive flexfields, but able to capture multiple sets of information within a context and multiple contexts grouped to appear in a named region of a user interface page. Some extensible flexfields allow grouping contexts into categories.

feature choice

A selection you make when configuring offerings that modifies a setup task list, or a setup page, or both.

fixed rate type

Rate you set between two currencies that remains constant. For example, a rate set between the euro currency and each Economic and Monetary Union (EMU) currency during the conversion to the euro currency.

flexfield

Grouping of extensible data fields called segments, where each segment is an attribute added to an entity for capturing additional information.

flexfield segment

An extensible data field that represents an attribute on an entity and captures a single atomic value corresponding to a predefined, single extension column in the Oracle Fusion Applications database. A segment appears globally or based on a context of other captured information.

free-form content type

A content type that contains a code, name, and description only, and does not contain any properties until you add it to a profile type.

function security

The control of access to a page or a specific widget or functionality within a page. Function security controls what a user can do.

global area

The region across the top of the user interface. It provides access to features and tools that are relevant to any page you are on.

grade

A component of the employment model that defines the level of compensation for a worker.

GTIN

Abbreviation for Global Trade Identification Number
**HCM**
Abbreviation for Human Capital Management.

**HCM data role**
A job role, such as benefits administrator, associated with specified instances of Oracle Fusion Human Capital Management (HCM) data, such as one or more positions or all persons in a department.

**HCM securing object**
An HCM object that secures access to both its own data and data in other, related objects. For example, access to a specified set of person records can allow access to data secured by person records, such as goal plans and evaluations.

**identity**
A person representing a worker, supplier, or customer.

**instance qualifier set**
A set of values that uniquely identifies multiple instances of the same profile item.

**interface table**
A database table used for transferring data between applications or from an external application or data file.

**inventory organization**
An organization that tracks inventory transactions and balances, and can manufacture or distribute products.

**item master**
A collection of data that describes items and their attributes recorded in a database file.

**item organization**
Item definition where inventory balances are not stored and movement of inventory is not tracked in the applications. Item attributes that carry financial and accounting information are hidden.

**Items**
Entries within the Product master database. For example, items for a manufacturing company can include nuts, bolts, and screws.

**job**
A generic role that is independent of any single department or location. For example, the jobs Manager and Consultant can occur in many departments.
job role
A role for a specific job consisting of duties, such as an accounts payable manager or application implementation consultant. A type of enterprise role.

key flexfield
Configurable key consisting of multiple parts or segments, each of which may be meaningful individually or in combination with the others. Key flexfields are commonly implemented to represent part numbers and account numbers.

key flexfield segment instance
A single occurrence of a key flexfield segment in a key flexfield structure instance.

key flexfield structure
The arrangement of segments in a key flexfield. In some cases, multiple structures can be defined for a single key flexfield.

key flexfield structure instance
A single occurrence of a key flexfield structure that shares the same order of segments as every other instance of the key flexfield structure, but uses different value sets to validate the segments.

lead rank
A configurable set of values such as hot, warm, or cool used to prioritize leads for lead qualification and sales engagement.

legal employer
A legal entity that employs people.

legal entity
An entity is identified and given rights and responsibilities under commercial law, through the registration with the country’s appropriate authority.

legal reporting unit
The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.

legislative data group
A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.
line of business
Set of one or more highly related products which service a particular customer transaction or business need. Refers to an internal corporate business unit.

lookup code
A value available for lookup within a lookup type such as the code BLUE within the lookup type COLORS.

lookup type
A set of lookup codes to be used together as a list of values on a field in the user interface.

mainline
A branch of data that serves as a single source of truth.

managed person
In Oracle Fusion Human Capital Management security, a person for whom the user can maintain some information. For example, line managers can maintain information about their direct and indirect reports, and workers can maintain information about themselves, their dependents, and their beneficiaries.

marketing activity
An instance of the execution of a campaign stage, such as delivering a specific treatment to a specific audience.

marketing list
A static selection of contacts for the purpose of communicating a marketing message by e-mail, direct mail or phone.

mini campaign
A simplified single stage communication platform that allows a marketer to interact with customers by e-mail or SMS.

model profile
A collection of the work requirements and required skills and qualifications of a workforce structure, such as a job or position.

multistage campaign
An integrated multichannel communication platform, that allows a marketer to achieve a specific marketing goal or objective through customer interaction, strategic advertisements and lead generation.

natural account
Categorizes account segment values by account type, asset, liability, expense, revenue, or equity, and sets posting, budgeting, and other options.
**offering**
A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.

**one click unsubscribe**
A marketing response form that can be inserted into an e-mail. E-mail recipients can click the link to remove themselves from the subscription list associated with a marketing treatment.

**OWLCS**
Abbreviation for Oracle WebLogic Communication Services. Offers the TPCC service to Fusion applications and sets up the calls via SIP integration with the telephony network.

**payroll statutory unit**
A legal entity registered to report payroll tax and social insurance. A legal employer can also be a payroll statutory unit, but a payroll statutory unit can represent multiple legal employers.

**pending worker**
A person who will be hired or start a contingent worker placement and for whom you create a person record that is effective before the hire or start date.

**person type**
A subcategory of a system person type, which the enterprise can define. Person type is specified for a person at the employment-terms or assignment level.

**personally identifiable information**
Any piece of information that can potentially be used to uniquely identify, contact, or locate a single person. Within the context of an enterprise, some PII data can be considered public, such as a person's name and work phone number, while other PII data is confidential, such as national identifier or passport number.

**PL/SQL**
Abbreviation for procedural structured queried language.

**position**
A specific occurrence of one job, fixed within one department, also often one location. For example, the position Finance Manager is an instance of the job Manager in the Finance Department.

**primary ledger**
Main record-keeping ledger.
**privilege**

A grant or entitlement of access to functions and data. A privilege is a single, real world action on a single business object.

**process**

A program that you schedule and run to process data and, if appropriate, generate output as a report. Also known as scheduled process.

**profile option**

User preferences and system configuration options consisting of a name and a value, that can be set at hierarchical levels of an enterprise. Also called a profile or user option.

**profile option level**

A level at which profile option values are defined. Site, product, and user are predefined levels.

**profile option level hierarchy**

The ordering of profile option levels. The order of the levels in the hierarchy determines which levels take precedence.

**profile option value**

The value portion of a profile option's name and value. A profile option may have multiple values set at different levels, such as site or user.

**profile type**

A template that defines the content sections of a profile, role access for each section, and whether the profile is for a person, or for a workforce structure such as a job or position.

**project expenditure organization**

An organization that can incur expenditures and hold financial plans for projects.

**PSTN**

Abbreviation for public switched telephone network which is the network of the world’s public circuit-switched telephone networks.

**public person**

In Oracle Fusion Human Capital Management security, a person for whom some basic information is publicly available. For example, users typically access the contact details of public persons, such as phone numbers and locations, using the person gallery.
qualified lead
A qualified lead is one where the lead qualification status has been updated to qualified. Generally, a lead is considered qualified and ready for conversion to a sale when the need, purchase interest, and budget are confirmed and a sales account and primary product are associated with the lead.

rating model
A scale used to measure the performance and proficiency of workers.

reference data
Data in application tables that is not transactional and not high-volume such as sales methods, transaction types, or payment terms, and can be shared and used across organizational boundaries.

reference data set
Contains reference data that can be shared across a number of business units or other determinant types. A set supports common administration of that reference data.

reference group
A logical grouping of tables that correspond to logical entities such as payment terms defined across multiple tables or views. Grouping establishes common partitioning requirements across the entities causing them to share the same set assignments.

request call back
A marketing response form that can be inserted into an e-mail. E-mail recipients can click the link to request a call back.

request unsubscribe from many
A marketing response form that can be inserted into an e-mail. E-mail recipients can click the link to remove themselves from the subscription list associated with a marketing treatment.

response
A recorded reaction of a prospect or customer to a marketing activity.

role
Controls access to application functions and data.

role hierarchy
Structure of roles to reflect an organization’s lines of authority and responsibility. In a role hierarchy, a parent role inherits all the entitlement of one or more child roles.
role mapping
A relationship between one or more job roles, abstract roles, and data roles and one or more conditions. Depending on role-mapping options, the role can be provisioned to or by users with at least one assignment that matches the conditions in the role mapping.

role provisioning
The automatic or manual allocation of an abstract role, a job role, or a data role to a user.

sales account
Parties with the usage Sales Account and a sales account profile containing sales information specific to the party. When a party has one sell to address, it ceases to be a sales prospect and becomes a new sales account. When the party purchases something, it changes from a new to an existing sales account.

sandbox
A runtime session that commits changes out of reach of mainline users.

security profile
A set of criteria that identifies one or more human capital management (HCM) objects of a single type for the purposes of securing access to those objects. Security profiles can be defined for persons, organizations, positions, countries, LDGs, document types, payrolls, and payroll flows.

security reference implementation
Predefined function and data security in Oracle Fusion Applications, including role based access control, and policies that protect functions, data, and segregation of duties. The reference implementation supports identity management, access provisioning, and security enforcement across the tools, data transformations, access methods, and the information life cycle of an enterprise.

segregation of duties
An internal control to prevent a single individual from performing two or more phases of a business transaction or operation that could result in fraud.

service provider model
A business unit that provides specific business functions for another business unit.

set
Reference data that is organized into groups appropriate to organizational entities, to enable reference data sharing.
set enabled
An entity, such as a lookup, customer, location, organization, or document attachment, that is allowed to participate in reference data sharing by drawing on the data of a reference data set.

SOA
Abbreviation for service-oriented architecture.

spot rate type
Rate you enter to perform conversion based on this rate as of a specific date. This rate applies to the immediate delivery of a currency.

SQL predicate
A type of condition using SQL to constrain the data secured by a data security policy.

stage
A unit of the marketing campaign design that contains a single purpose that contributes to the overall objective of the campaign. The stage purpose can be for interacting, events, advertising, and generating leads.

subscribe to list
A marketing response form that can be inserted into an e-mail. E-mail recipients can click the link to add themselves to the subscription list associated with a marketing treatment.

system person type
A fixed name that the application uses to identify a group of people.

territory
A legally distinct region that is used in the country field of an address.

territory owner
Resource assigned to manage a territory and is typically accountable for the work objects, such as opportunities, that are within the boundaries of the territory.

territory proposal
A sandbox container used to model territory changes. All valid territories within a proposal become active on the proposal activation date.

treatment
A marketing term that represents a combination of a marketing message for targeted audience members and delivery options for third party suppliers.
**tree**

Information or data organized for display into a hierarchy with one or more root nodes connected to branches of nodes. Each node corresponds to data from one or more data sources. A tree must have a structure.

**tree structure**

Characteristics applied to trees, such as what data to include or how the tree is versioned and accessed.

**tree version**

An instance of a tree. If a tree is associated with a reference data set, all versions belong to one set. Includes life cycle elements such as start and end date and a status indicator whether the tree is active or not.

**unsubscribe from list**

A marketing response form that can be inserted into an e-mail. E-mail recipients can click the link to remove themselves from the subscription list associated with a marketing treatment.

**user rate type**

Rate you enter at journal entry time to convert foreign currency transactions to your functional currency.

**value set**

A set of valid values against which values entered by an end user are validated. The set may be tree structured (hierarchical).

**work relationship**

An association between a person and a legal employer, where the worker type determines whether the relationship is a nonworker, contingent worker, or employee work relationship.

**workflow**

An automated process in which tasks are passed from a user, a group of users, or the application to another for consideration or action. The tasks are routed in a logical sequence to achieve an end result.

**XML filter**

A type of condition using XML to constrain the data secured by a data security policy.