Oracle Fusion Marketing
Implementing Marketing

Release 8

April 2014
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 ................................................ 2-10
FAQs for Currencies and Currency Rates ............................................. 2-16
Define Enterprise Structures for Marketing ......................................... 2-17
Define Initial Configuration ................................................................. 2-25
FAQs for Define Initial Configuration ................................................. 2-45
Define Enterprise for Marketing ......................................................... 2-45
FAQs for Define Enterprise for Marketing ........................................... 2-50
Define Geographies ............................................................................. 2-51
FAQs for Define Geographies .............................................................. 2-59
Define File-Based Data Import ............................................................ 2-60
FAQs for File-Based Data Import ....................................................... 2-61
Define Legal Entities for Marketing ................................................... 2-61
FAQs for Define Legal Entities for Marketing ....................................... 2-71
Define Business Units for Marketing .................................................. 2-71
Define Workforce Structures for CRM .............................................. 2-75
FAQs for Define Workforce Structures for CRM ................................. 2-83
Define Facilities for Marketing .......................................................... 2-85
Define Social Networking ................................................................. 2-89
FAQs for Define Social Networking .................................................... 2-108
Define Security for Customer Relationship Management .................... 2-108
Define Users for CRM ............................................................... 2-128
FAQs for Define Users for CRM ....................................................... 2-150
Define Automated Governance, Risk, and Performance Controls ........ 2-151
Define Approval Management for Customer Relationship Management ... 2-163
Define Help Configuration ............................................................... 2-164
FAQs for Define Help Configuration .................................................. 2-167
Define Application Toolkit Configuration ......................................... 2-168
Map Reports to Work Areas ............................................................... 2-169
FAQs for Map Reports to Work Areas ................................................ 2-169
Set Watchlist Options ...................................................................... 2-170

3 Maintain Common Reference Objects
4 Define Applications Core Configuration

Maintain Common Reference Objects: Overview ............................................ 3-1
Define Application Taxonomy ......................................................................... 3-1
Define Reference Data Sharing ........................................................................ 3-4
Define ISO Reference Data ............................................................................. 3-8
Manage Data Security Policies ................................................................. 3-12
Set Activity Stream Options ........................................................................ 3-22
Manage Menu Customizations ..................................................................... 3-22
Manage Audit Policies .................................................................................. 3-28
Manage Oracle Social Network Objects .................................................. 3-30
Manage Applications Core Common Reference Objects ................................ 3-36

5 Define Common CRM Configuration

Define Source Systems .................................................................................. 5-1
FAQs for Source Systems ............................................................................. 5-2
Manage Public Unique Identifier Profile Options ........................................ 5-2
Define Party Usages ..................................................................................... 5-4
FAQs for Define Party Usages ....................................................................... 5-5
Define Names ................................................................................................ 5-5
Define Identifiers .......................................................................................... 5-6
Define Relationships .................................................................................... 5-6
Define Classifications .................................................................................. 5-10
FAQs for Define Classifications ................................................................... 5-12
Define Contact Points .................................................................................. 5-12
FAQs for Define Contact Points ................................................................... 5-14
Define Data Formats ..................................................................................... 5-14
FAQs for Define Data Formats ...................................................................... 5-19
Define Resource Information ......................................................................... 5-20
FAQs for Manage Resources ......................................................................... 5-22
Define Resource Organization Information .................................................. 5-23
FAQs for Define Resource Organization Information ..................................... 5-28
Define Resource Role Information ................................................................ 5-30
FAQs for Define Resource Role Information ............................................... 5-32
Define Resource Team Information ............................................................. 5-33
FAQs for Define Resource Team Information ............................................... 5-33
Define Products: Define Basic Items ............................................................ 5-34
FAQs for Define Basic Items ......................................................................... 5-39
Manage Units of Measure ............................................................................... 5-39
FAQs for Units of Measure ............................................................................. 5-41
Define Products: Define Advanced Items ................................................................. 5-42
FAQs for Define Advanced Items ............................................................................. 5-46
Define Sales Catalogs .............................................................................................. 5-46
FAQs for Define Sales Catalogs .............................................................................. 5-55
Manage Bulk Data Export ...................................................................................... 5-58
FAQs for Manage Bulk Data Export ...................................................................... 5-62
Manage Accounting Calendar ............................................................................... 5-63
Define Sales Prediction Configuration ................................................................. 5-64
Define Outlook Integration .................................................................................... 5-74
FAQs for Define Outlook Integration ..................................................................... 5-92
Set Up Customer Center ........................................................................................ 5-93
Manage Customer Center Trees .......................................................................... 5-95
FAQs for Manage Customer Center Trees ............................................................. 5-98
Manage Customer Center Assessment Template ................................................. 5-98
FAQs for Manage Customer Center Assessment Template .................................... 5-104
Manage Customer Center Task Template ............................................................ 5-104
FAQs for Manage Customer Center Task Template .............................................. 5-106
Define Enhanced Click-to-Dial .............................................................................. 5-107

6 Assignment Manager

Manage Assignment Manager Objects .................................................................. 6-1
Manage Assignment Manager Mappings ............................................................... 6-8
Manage Assignment Manager Rules ...................................................................... 6-11
FAQs for Manage Assignment Manager Objects .................................................. 6-17
Sales Account and Territory Assignment ............................................................. 6-19
FAQs for Sales Account and Territory Assignment ............................................... 6-23
Manage Sales Lead Assignment Objects .............................................................. 6-25
Running Assignment Processes ............................................................................ 6-34
Running Assignment Manager Reports ................................................................. 6-36

7 Define Territory Management Configuration

Territory Components: How They Work Together ................................................... 7-1
Modeling Territories: Things to Consider ............................................................. 7-2
Define Territory Management Dimensions ......................................................... 7-3
Territory Export and Import ................................................................................... 7-15

8 Define Lead Management

How Lead Components Fit Together ................................................................ ..... 8-1
Implementing Lead Management: Critical Choices ............................................... 8-3
Configure Assignment Manager for Lead Processing ............................................. 8-5
Configure Sales Lead Assignment Rules ............................................................... 8-8
Configure Assessment Reference Data for Sales Leads ......................................... 8-9
FAQs for Configure Assessment Reference Data for Sales Leads ............................ 8-17
Configure Notes for Sales Leads ........................................................................... 8-18
Configure Tasks for Sales Leads ......................................................................... 8-18
Configure Sales Leads .......................................................................................................................... 8-20

9 Define Common Marketing Configuration

Manage Marketing Standard URLs ........................................................................................................ 9-1
Review Diagnostics for Marketing .......................................................................................................... 9-4
Define Marketing Promotions ................................................................................................................. 9-5
Define Campaign and Response Notes for Marketing ........................................................................... 9-6
Define Campaign Tasks for Marketing ................................................................................................... 9-7
Define Marketing Templates: Manage Marketing Campaign Templates ................................................. 9-9
FAQs for Manage Marketing Campaign Templates .................................................................................. 9-17
Define Marketing Templates: Manage Marketing Treatment Templates ............................................... 9-19
Configure Marketing Campaigns and Responses ................................................................................... 9-22
Manage Social Marketing Connectors .................................................................................................... 9-25

10 Define Campaign Fulfillment System

Importing Marketing Suppliers: Explained .............................................................................................. 10-1

11 Define E-Mail Server

E-Mail Architecture for Marketing: How It Fits Together ......................................................................... 11-1
Manage Marketing E-Mail Sending Daemon: Explained ......................................................................... 11-3

12 Define Segmentation Manager

Oracle Fusion Marketing Segmentation: Architectural Overview ........................................................... 12-1
Oracle Fusion Marketing Segmentation: Implementation Planning ........................................................ 12-2
Marketing Segmentation Administration: Highlights ............................................................................... 12-3
Marketing Segments and Segment Trees: Highlights ............................................................................. 12-5
Configuring Custom Subject Areas for Segmentation: Points to Consider ........................................... 12-6

13 Define Transactional Business Intelligence Configuration

Define Transactional Business Intelligence Configuration: Highlights .................................................... 13-1
Manage Security for Transactional Business Intelligence ....................................................................... 13-1

14 Define Extensions for Marketing

Define Custom Enterprise Scheduler Jobs .............................................................................................. 14-1
Contextual Addresses .............................................................................................................................. 14-4
Customizations and Sandboxes .............................................................................................................. 14-4
Exporting and Importing Supported CRM Customizations: Explained .............................................. 14-6
Marketing Customizations ...................................................................................................................... 14-8
Customizing Marketing Campaign Stage Pages Using Application Composer: Explained ................. 14-11
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.

Note

If you don’t see any help icons on your page, then click the Show Help icon button in the global area. However, not all pages have help icons.

You can add custom help files to replace or supplement the provided content. Each release update includes new help content to ensure you have access to the latest information. Patching does not affect your custom help 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. To access the guides, go to any page in Oracle Fusion Applications Help and select Documentation Library from the Navigator menu.

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.

These guides cover specific business processes and offerings. Common areas are addressed in the guides listed in the following table.

<table>
<thead>
<tr>
<th>Guide</th>
<th>Intended Audience</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common User Guide</td>
<td>All users</td>
<td>Explains tasks performed by most users.</td>
</tr>
<tr>
<td>Common Implementation Guide</td>
<td>Implementors</td>
<td>Explains tasks within the Define Common Applications Configuration task list, which is included in all offerings.</td>
</tr>
<tr>
<td>Functional Setup Manager User Guide</td>
<td>Implementors</td>
<td>Explains how to use Oracle Fusion Functional Setup Manager to plan, manage, and track your implementation projects, migrate setup data, and validate implementations.</td>
</tr>
<tr>
<td>Technical Guides</td>
<td>System administrators, application developers, and technical members of implementation teams</td>
<td>Explain how to install, patch, administer, and customize Oracle Fusion Applications.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note Limited content applicable to Oracle Cloud implementations.</td>
</tr>
</tbody>
</table>

For other guides, go to Oracle Technology Network at http://www.oracle.com/technetwork/indexes/documentation.

**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 details on 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 Oracle Enterprise Repository at http://fusionappsoer.oracle.com 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.

- Other technical information such as reusable components, policies, architecture diagrams, and topology diagrams.

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_grp@oracle.com. You can use Send Feedback to Oracle from the Settings and Actions menu in 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 Sales Cloud 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 for comparable Oracle Fusion applications 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.
• Achieve 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.

The following documentation resources are available for learning how to configure Oracle Fusion Applications.
• Functional Setup Manager Developer's Guide
• Common Implementation Guide
• Customer Data Management Implementation Guide
• Enterprise Contracts Implementation Guide
• Marketing Implementation Guide
• Sales Implementation Guide
• Fusion Accounting Hub Implementation Guide
• Financials Implementation Guide
• Compensation Management Implementation Guide
• Workforce Deployment Implementation Guide
• Workforce Development Implementation Guide
• Incentive Compensation Implementation Guide
• Procurement Implementation Guide
• P6 EPPM Administrator's Guide for an Oracle Database
• P6 EPPM Administrator's Guide for Microsoft SQL Server Database

Implementing Oracle Fusion CRM with the Functional Setup Manager:
Functional Overview

Oracle Fusion Applications use the Setup and Maintenance work area, available in the Navigator menu under the Tools heading, as the central work area for planning and executing the setup tasks required to implement and maintain your Oracle Fusion CRM application.

Every phase of application implementation lifecycle is supported by the Setup and Maintenance work area, also referred to as the Functional Setup Manager.

Using the list of tasks on the left side of the Setup and Maintenance work area, you can:
• Review the functionality available for implementation using the Getting Started task.
• Select what functionality you want to implement using the Configure Offerings task.
• Create and manage implementation projects using the Manage Implementation Projects task.

You can perform individual setup and maintenance tasks, either from an implementation project or by searching for the task or task group in the All Tasks tab in the work area.

Although you are not required to create an implementation project for setting up your application functionality, implementation projects make it possible
to export your setups for use in other environments using the Manage Configuration Packages task. Exporting is useful if you want to move your setups from a test environment to a production environment, for example.

**Reviewing Functionality Available for Implementation**

Using the Getting Started task, you can review the functionality available for you to implement.

Oracle Fusion Applications functionality is organized by offerings. Offerings comprise tasks related to business processes that are typically provisioned and implemented as a unit. For Oracle Fusion CRM, there are five relevant offerings:

<table>
<thead>
<tr>
<th>Offering</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Contracts</td>
<td>Oracle Fusion Enterprise Contracts provides the most complete solution for managing sales, procurement, and other contracts. It is a comprehensive offering for standardizing corporate contract policies, improving internal controls, and complying with all contractual obligations and regulatory requirements.</td>
</tr>
<tr>
<td>Customer Data Management</td>
<td>Configure everything you need to manage customers, contacts, and resources, and ensure data quality.</td>
</tr>
<tr>
<td>Incentive Compensation</td>
<td>Configure how you manage incentive compensation, including incentive plans, performance monitoring, credit and rollup rules, earnings calculations, and payment determination.</td>
</tr>
<tr>
<td>Marketing</td>
<td>Configure everything you need to manage marketing campaigns and leads, including segmentation and e-mail servers.</td>
</tr>
<tr>
<td>Sales</td>
<td>Configure everything you need to manage sales planning and forecasting, pursue and track opportunities, align territories, and distribute sales quotas.</td>
</tr>
</tbody>
</table>

Click an offering to obtain details in a series of reports. Available reports include:

- Offering Content Guide
  - List of business processes and activities covered by the offering.
- Associated Features
  - Lists the options and features available for implementation.
- Setup Task Lists and Tasks
  - Complete list of implementation tasks organized in task lists. The tasks, which include all the common and prerequisite tasks, are organized in the order they are to be implemented.
- Related Business Objects
  - Lists business objects along with the applicable Web service used for the export and import process. The report displays the business object, associated product and enterprise application, and the related Web service.
• Related Enterprise Applications
  List of Java Enterprise Edition applications used for entering, exporting, and importing setup data.

Selecting What Functionality You Want to Implement

Using the Configure Offerings task, you can select which optional functionality or features will be available for implementation within each offering.

For example, for the Sales offering, you can enable for implementation the following optional functionality:

• Competitors
• Quotas
• References
• Sales Catalog
• Sales Prediction Engine
• Outlook Integration
• Territory Management
• Lead Management
• Sales Business Intelligence Analytics
• Partner Management

On the Configure Offerings page, you can also select optional features such as implementing a local help system, or automatic phone dialing with Enhanced Click-to-Dial.

Creating and Managing Implementation Projects

When you are ready to implement, you can use the Manage Implementation Projects task to create the list of setup tasks you will need from one or more offerings. While you can implement your application by searching for each setup task individually, creating an implementation project provides you with many project management features, such as assigning tasks and tracking their completion. Creating an implementation project also makes it possible for you to export your setup after it is complete so you can reuse it in a different environment.

Although there are no restrictions on the number of offerings you can include in a single implementation project, you should create separate projects for each offering:

• If you are implementing and deploying offerings at different times
• To keep the list of setup tasks manageable within a project

About Implementation Tasks and Task Lists

The setup tasks in an offering or in an implementation project are organized in nested task lists in the suggested implementation order:

1. The most common setup tasks which apply across offerings
2. Tasks common to an application area such as CRM
3. Tasks common across multiple modules in an application area
4. Tasks for specific modules such as Opportunity Management, Territory Management, and Sales Forecasting
Note that:

- You do not have to complete all of the setup tasks in a project or an offering. Many setup tasks are optional or have default values already populated for you.
- Each offering is self-contained and includes the common tasks. If you are implementing multiple offerings, you do not have to duplicate the common task setup.

You can obtain more information about many setup task lists or individual tasks by clicking the question mark help icon. Compilations of the all the implementation help topics are available in implementation guides.

**Implementing Oracle Fusion CRM with the Functional Setup Manager: Special Considerations**

This topic highlights the differences between using the Functional Setup Manager to implement Oracle Fusion CRM on premise and in the cloud.

While there is no difference in how you use the Functional Setup Manager on premise and in the cloud, there is a difference in the setup tasks you carry out to implement your application. When you implement Oracle Sales and Marketing Cloud Service:

- Some required setup tasks, listed in the following table, are already completed for you based on the information you provided on the initial questionnaire.
- Some tasks in an offering may not be available. For example, Oracle Sales and Marketing Cloud Service permits you to import customer and other application data from a file, but does not permit you to load data directly into the application using a loader. Some security customizations, such as the creation of additional enterprise roles, must be done for you by service administrators. Details about the availability of specific tasks are provided in related topics.

The following table lists tasks set up for you when you implement in the Oracle Sales and Marketing Cloud Service.

<table>
<thead>
<tr>
<th>Implementation Task Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage Enterprise HCM Information</td>
<td>Oracle enters the company name you provided. You can search for the name from the Manage Enterprise HCM Information page and edit it. All of the other fields on the Edit Enterprise page are for Oracle Fusion Human Capital Management (HCM), and so they are not relevant for CRM.</td>
</tr>
<tr>
<td>Manage Legal Address</td>
<td>This is the address you provided. As a best practice, use the same legal corporate address used for tax information.</td>
</tr>
<tr>
<td>Manage Legislative Data Group</td>
<td>The legislative data group is required for creating a legal employer. Legislative data groups are used for partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. CRM-only implementations require only one data group.</td>
</tr>
</tbody>
</table>
Manage Legal Entity

The legal entity is required for creating employee users. When you create a user who is an employee, you must select the legal entity, the legal employer. You can set up multiple legal entities if your business requires them, but, for CRM-only implementations, you can assign all employees to just one corporate legal employer. You must complete this task before executing the next step: Manage Legal Entity HCM Information task.

Manage Legal Entity HCM Information

In this task, you associate the legislative data group defined in the Manage Legislative Data Group task with the Legal Entity HCM Information.

Manage Business Unit

Oracle creates one business unit for creating your employee users. The name of the business unit is based on your company name. Because all CRM transactions occur in one business unit, you can assign all employees to the same corporate business unit.

Select the Manage Users link from the Navigator.

Oracle creates one initial user for you. Sign in as the initial user to create other users, including other users with the permissions to perform implementation tasks. These users are called setup users.

For step-by-step instructions on creating a setup user and a CRM application user, see the topics Creating Setup Users for the Oracle Sales and Marketing Cloud Service: Worked Example.

### Implementation Tasks Completed for You in the Oracle Sales Cloud Service: Explained

Oracle completes the implementation tasks listed in this topic according to the information you provided when you signed up for the Oracle Sales Cloud Service. You can verify or edit the values for each task.

The following table lists the completed tasks in the order they are implemented. Except for the Manage Users task, you can review each setup task and make any necessary changes by searching for the task name in the Setup and Maintenance work area and clicking the Go to Task button.

Links to both the Setup and Maintenance work area and the Manage Users task appear in the Navigator menu at the top of any application page.

<table>
<thead>
<tr>
<th>Implementation Task Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage Enterprise HCM Information</td>
<td>Oracle enters the company name you provided. You can search for the name from the Manage Enterprise HCM Information page and edit it. All of the other fields on the Edit Enterprise page are for Oracle Fusion Human Capital Management (HCM), and so they are not relevant for CRM.</td>
</tr>
<tr>
<td>Manage Legal Address</td>
<td>This is the address you provided. As a best practice, use the same legal corporate address used for tax information.</td>
</tr>
<tr>
<td>Task Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Manage Legislative Data Group</td>
<td>The legislative data group is required for creating a legal employer. Legislative data groups are used for partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. CRM-only implementations require only one data group.</td>
</tr>
<tr>
<td>Manage Legal Entity</td>
<td>The legal entity is required for creating employee users. When you create a user who is an employee, you must select the legal entity, the legal employer. You can set up multiple legal entities if your business requires them, but, for CRM-only implementations, you can assign all employees to just one corporate legal employer. You must complete this task before executing the next step: Manage Legal Entity HCM Information task.</td>
</tr>
<tr>
<td>Manage Legal Entity HCM Information</td>
<td>In this task, you associate the legislative data group defined in the Manage Legislative Data Group task with the Legal Entity HCM Information.</td>
</tr>
<tr>
<td>Manage Business Unit</td>
<td>Oracle creates one business unit for creating your employee users. The name of the business unit is based on your company name. Because all CRM transactions occur in one business unit, you can assign all employees to the same corporate business unit.</td>
</tr>
<tr>
<td>Select the Manage Users link from the Navigator</td>
<td>Oracle creates one initial user for you. Sign in as the initial user to create other users, including other users with the permissions to perform implementation tasks. These users are called setup users. For step-by-step instructions on creating a setup user and a CRM application user, see the topics Creating Setup Users for the Oracle Sales Cloud Service: Worked Example.</td>
</tr>
</tbody>
</table>

**Implementation Projects: Explained**

An implementation project is the list of setup tasks you need to complete to implement selected offerings and functional areas. You create a project by selecting the offerings and functional areas 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 functional areas that are configured for implementation. Implementation managers specify which of those offerings and functional areas 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 functional area, and one or more or features.

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, functional areas 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 functional areas 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 functional areas 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 Sales Cloud 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 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.

**Note**
For security and audit best practice, implementation users have person records and appropriate role-based security access. So that appropriate roles can be assigned to implementation users, you must run the process Retrieve Latest LDAP Changes before you create implementation users.

During initial implementation, the installation super user performs the task Run User and Role Synchronization Process to run the Retrieve Latest LDAP Changes process.

**Tip**
The user name and password of the installation super user are created during installation provisioning of Oracle Fusion Applications. For details of the user name and password, contact your system administrator or the person who installed Oracle Fusion Applications.

**Define Implementation Users**

**Getting Started with an Implementation: Overview**

To start an Oracle Fusion Applications implementation, you must set up one or more initial users using the super user that was created during installation and provisioning of the Oracle Fusion Applications environment, or using the initial administrator user provided by Oracle for Oracle Cloud Application Services implementations. Because Oracle Fusion Applications is secure as delivered, the process of enabling the necessary setup access for initial users requires several specialized steps when getting started with an implementation.

The following high level steps are required for starting an implementation.

1. If you are not starting an Oracle Cloud Application Services implementation, sign into Oracle Identity Manager (OIM) as the OIM Administration users and provision the IT Security Manager job role with roles for user and role management. This enables the super user account,
which is provisioned with the IT Security Manager job role, to create implementation users.

2. For starting all implementations, sign in as the user with initial access: either the Oracle Fusion Applications installation super user or the initial Oracle Cloud Application Services administrator user.

3. Select an offering to implement, and generate the setup tasks needed to implement the offering.

4. Perform the following security tasks:
   b. Create an IT security manager user by using the Create Implementation Users task.
   c. Provision the IT security manager with the IT Security Manager role by using the Provision Roles to Implementation Users task.

5. As the newly created IT security manager user, sign in to Oracle Fusion Applications and set up at least one implementation user for setting up enterprise structures.
   a. Create an implementation user by using the Create Implementation Users task.
   b. Provision the implementation user with the Application Implementation Manager job role or the Application Implementation Consultant job role by using the Provision Roles to Implementation Users task. The Application Implementation Consultant job role inherits from all product-specific application administrators and entitles the necessary View All access to all secured objects.
   c. Optionally, create a data role for an implementation user who needs only the limited access of a product-specific Application Administrator by using the Create Data Role for Implementation Users. Then assign the resulting data role to the implementation user by using the Provision Roles to Implementation Users task.

The figure shows the task flow from provisioning the IT Security Manager job role with the user and role management entitlement to creating and provisioning implementation users for enterprise setup.
Initial Security Administration: Critical Choices

After installation and provisioning, and before setting up enterprise structures and implementing projects, you must establish required entitlement for the super user account and at least one implementation user to proceed with the implementation. Once initial enterprise structure setup is complete, additional users may be created through processes available in Human Capital Management (HCM).

Initial security administration consists of the following.

- Preparing the IT Security Manager job role
- Synchronizing users and roles from Lightweight Directory Access Protocol (LDAP) with HCM
- Creating implementation users
- Optionally creating data roles for implementation users
- Provisioning implementation users with roles

Once the first implementation project begins and the enterprise work structure is set up, use standard user and security management processes such as the Manage Users task to create and manage additional users. Do not use the Create Implementation Users task after your enterprise has been set up.
Preparing the IT Security Manager Job Role

Initially the super user is not provisioned to manage users and roles.

You must add the following Oracle Identity Management (OIM) roles to the IT Security Manager job role’s role hierarchy to enable the super user to create one or more initial implementation users.

- Identity User Administrators
- Role Administrators

Additionally, you must assign the Xellerate Users organization to the IT Security Manager role.

Synchronizing Users and Roles from LDAP

After configuring an offering and setting up the task lists for implementation, the Run User and Roles Synchronization Process task is available to the super user for synchronizing users and roles in the LDAP store with Oracle Fusion Human Capital Management (HCM).

Defining Initial Implementation Users

The super user is provisioned with roles that provide broad access to Oracle Fusion Middleware and Oracle Fusion Applications administration, and is not suitable as an implementation user in most enterprises. The super user should define at least one implementation user, which consists of creating the user account and provisioning it with at least the Application Implementation Consultant and Application Implementation Manager job roles.

As a security guideline, define an IT security manager user who in turn defines one or more implementation users to set up enterprise structures. The IT security manager users can provision the implementation user with the Application Implementation Consultant role, which entitles access to all enterprise structures. Or the IT security manager can create a data role that restricts access to enterprise structures of a specific product and provisioning that role.

Depending on the size of your implementation team, you may only need a single implementation user for security administration, implementation project management, enterprise structures setup, and application implementation. That single user must then be provisioned with all indicated roles, and therefore broad access.

Creating Implementation Users

The super user creates one or more implementation users by performing the Create Implementation Users task.

Note

This initial implementation user is a user account created in Oracle Identity Management only, specifically for setting up enterprise structures, and is not related to a real person or identity such as a user defined in HCM.
Creating Data Roles for Implementation Users

As an alternative to provisioning an implementation user with the Application Implementation Consultant role to access all enterprise structures, you may need implementation users with access restricted to enterprise structures for specific products. In this case, use the Create Data Roles for Implementation Users task to create a data role based on a job role with less broad access, such as the HCM Application Administrator job role.

Provisioning Roles to Implementation Users

After creating an implementation user, you must provision the user with one or more roles by performing the Provision Roles to Implementation Users task.

For example, assign a role to the implementation user that provides the access necessary for setting up the enterprise. Depending on need, provision to the implementation user the predefined Applications Implementation Consultant role or a product family-specific administrator data role, such as a data role based on the predefined Financials Applications Administrator.

Caution

The Application Implementation Consultant has broad access. It is a very useful role for experimentation or setting up a pilot environment, but may not be suitable for implementation users in a full implementation project.

Initial Security Administration: Worked Example

This example illustrates initial security administration after having installed and provisioned an Oracle Fusion Applications environment.

In Oracle Fusion Applications, you manage users and security through Oracle Fusion Human Capital Management (HCM) user management flows, which are included in each of the offering task lists. However, the HCM task flows require that enterprise structures have been set up, and yet to add users who can set up enterprise structures you need to have set up HCM. Therefore, you need to create one or more initial implementation users who are responsible for providing the following.

- Users and their applications security management
- Implementation project management
- Initial enterprise structures management

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decision</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>How to sign in to Oracle Fusion Applications for the first time</td>
<td>Use the super user account that was created when installing and provisioning Oracle Fusion Applications (for example, FAADMIN).</td>
</tr>
</tbody>
</table>
How to ensure that the roles and users in the Lightweight Directory Access Protocol (LDAP) store match what is available for selection when defining implementation users

Perform the Run User and Roles Synchronization Process task.

--

How to create a first implementation user

Prepare the IT Security Manager job role for user and role management so the super user and any other user provisioned with the IT Security Manager job role can manage users and roles.

--

How to establish security administration users

Define an IT security manager user provisioned with the IT Security Manager job role.

--

How to establish an implementation user with access to set up enterprise structures

Define an implementation user provisioned with the Application Implementation Consultant job role.

---

You create an initial implementation user by performing the following tasks.

1. The Oracle Identity Management System Administrator user provisions the IT Security Manager job role with roles for user and role management.

2. The Oracle Fusion Applications super user synchronizes LDAP users with HCM user management so that users can be provisioned with roles through HCM.

3. The Oracle Fusion Applications super user performs the Create Implementation Users task to create one or more IT security manager and administrator users provisioned with security administrative entitlement.

4. The IT Security Manager user signs in to Oracle Fusion Applications and performs the Create Implementation Users task to create implementation managers and users.

5. The IT Security Manager user provisions implementation users for enterprise structure setup.

---

Note

The following tasks assume that the super user has configured an offering and set up task lists. When not following a task flow within an activity, you can find tasks in Navigator > Tools > Setup and Maintenance > All Tasks. Search for the task and click its Go to Task icon in the search results.

---

Preparing the IT Security Manager Role

The super user that was created when installing and provisioning Oracle Fusion Applications (for example, FAADMIN), or the initial administrator user provided by Oracle for Oracle Cloud Application Services, has all necessary access for implementing Oracle Fusion Applications and administering security. This access is provided by the following roles:

- Application Implementation Consultant
- IT Security Manager

Neither of these roles provides access needed for creating and managing Oracle Fusion Applications users. Therefore, you must add the following two OIM roles to the IT Security Manager role:

- Identity User Administrators
• Role Administrators

The following procedure is prerequisite to an IT security manager or administrator creating an initial one or more implementation users.

1. While signed into Oracle Identity Manager as the OIM System Administrator user, click the Administration link in the upper right of the Oracle Identity Manager.

   This accesses the Welcome to Identity Manager Delegated Administration menu.

2. In the Roles list of tasks, click Advanced Search - Roles. Search for the Identity Users Administrators role by entering the role name in Display Name and clicking Search.

   In the Search Results, click the role’s Display Name.

3. On the Hierarchy tab, select Inherits From and click Add.

4. In the Add Parent Role to: IDENTITY USER ADMINISTRATORS window, select the role category: Common - Job Roles and add the IT Security Manager.

   Click the arrow icon to show the list of available roles. Select IT Security Manager and move it to the Roles to Add list. Click Save.

5. Search for the Role Administrators role, and repeat steps 1 to 4 to add that role to the IT Security Manager role’s role inheritance.

6. Assign the IT Security Manager role to the Xellerate Users organization.

   a. In the Welcome to Identity Manager Delegated Administration menu (see step 1, above), in the Organizations list of tasks, click Advanced Search - Organizations.

   b. Search for the Xellerate Users organization by entering Xellerate Users in Display Name and clicking Search.

   c. In the Search Results, click the organization’s Display Name. The Xellerate Users page appears.

   d. Click the Administrative Roles link in the row of links above the Xellerate Users.

   e. In Filter By Role Name of the Details window, enter the following string:

      *IT_SECURITY_MANAGER*

      Click Find.

   f. Enable Read, Write, Delete, and Assign.

   g. Click Assign.

   h. Click Confirm.

**Synchronizing Users and Roles from LDAP**

Lightweight Directory Access Protocol (LDAP) must be synchronized with HCM user management so that users can be provisioned with roles through HCM.
1. Sign in to Oracle Fusion Applications using the super user’s user name (for example FAADMIN) and password.

   If you do not know the super user name and password, check with your system administrator or the person who installed Oracle Fusion Applications. For more information about account creation in Oracle Fusion Applications provisioning, see the Oracle Fusion Applications Installation Guide.

2. Perform the Run User and Roles Synchronization Process task by clicking **Submit** in the Process Details page.

   The Retrieve Latest LDAP Changes process takes some time to complete the first time it is run.

3. Monitor completion of the Retrieve Latest LDAP Changes process from **Navigator > Tools > Scheduled Processes** before continuing with creating implementation users.

**Defining an IT Security Manager User**

The super user has broad access to Oracle Fusion Middleware and Oracle Fusion Applications administration. Due to this broad access, your enterprise needs users dedicated to managing users and applications security, such as an IT security manager user.

1. While signed in as the Oracle Fusion Applications super user, access the Create Implementation Users task and create an IT security manager.

   The Oracle Identity Manager appears.

2. Click **Create User**.

   For details, see the Creating Users section in the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

3. Provide the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last name</td>
<td>&lt;any valid string&gt;</td>
<td>Smith</td>
</tr>
<tr>
<td>Organization</td>
<td>Xellerate Users</td>
<td>N/A</td>
</tr>
<tr>
<td>User type</td>
<td>Non Worker</td>
<td>N/A</td>
</tr>
<tr>
<td>User login</td>
<td>&lt;any valid string&gt;</td>
<td>IT_SECURITY_MANAGER</td>
</tr>
<tr>
<td>Login password</td>
<td>&lt;any valid string&gt;</td>
<td>SeKur1TyPa$$w0Rd</td>
</tr>
</tbody>
</table>

   **Note**

   In Oracle Fusion Applications, an implementation user is a user account created in OIM only, specifically for implementation tasks, and is not related to a real person or identity such as a user defined in HCM.

4. Click **Save**.

5. On the Roles tab in the IT_SECURITY_MANAGER user creation task flow, click **Assign**.
6. In the Add Role window, search for the IT Security Manager role and click Add.

Defining an Implementation User for Enterprise Structures Setup

1. Sign in to Oracle Fusion Applications using the IT security manager user's name and password.

2. Create and provision an implementation user using the same task flow as for creating the IT security manager user in the previous section, except provision the following roles.
   - Application Implementation Manager
   - Application Implementation Consultant

Note
For an implementation to begin, at least one user must be provisioned with the Application Implementation Manager role, and another or the same user must be provisioned with the Application Implementation Consultant role. The Application Implementation Consultant has broad access to set up all enterprise structures.

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.

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, and 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 process 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 process 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
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.

Entering Daily Rates Manually: Worked Example

You are required to enter the daily rates for currency conversion from Great Britain pounds sterling (GBP) to United States dollars (USD) each day for your company InFusion America Inc.

Oracle Application Development Framework (ADF) Desktop Integration is an Excel add-in that must be loaded onto each client. Because ADF Desktop Integration is an add-in to Microsoft Office products, you can use this feature only if they have Microsoft Excel 2007 or above, Internet Explorer 7 or above,
and Microsoft Windows 7, XP Professional SP2, or Vista. Users must download the installation files from Navigator - Tools - Download Desktop Integrator Installer.

**Entering Daily Rates**

1. Navigate to the Period Close work area.
   Use the Period Close work area to link to close processes and currency process.

2. Click the Manage Currency Rates link.
   Use the Currency Rates Manager page to create, edit, and review currency rate types, daily rates, and historical rates.

3. Click the Daily Rates tab.
   Use the Daily Rates tab to review and enter currency rates.

4. Click the Create in Spreadsheet button.
   Use the Create Daily Rates spreadsheet to enter daily rates in a template that you can save and reuse.

5. Click in the From Currency field. Select the GBP - Pound Sterling list item.

6. Click in the To Currency field. Select the USD - US Dollar list item.

7. Click in the Conversion Rate field. Select the Spot list item

8. Click in the From Conversion field. Enter the desired information into the From Conversion field. Enter a valid value e.g. "8/1/2011".

9. Click in the To Conversion Date field. Enter the desired information into the To Conversion Date field. Enter a valid value e.g. "8/1/2011".

10. Click in the Conversion Rate field. Enter the desired information into the Conversion Rate field. Enter a valid value e.g. "1.33225".

11. Click the Submit button. Click the OK button twice.

12. Review the Record Status column to verify that all rows were loaded successfully.

13. Save the template to use to enter daily rates frequently. You can save the spreadsheet to either a local drive or a shared network drive.

**Updating Currency Rates: Worked Example**

You are required to change today’s daily rates that were already entered. The rates you are changing are for currency conversion from Great Britain pounds sterling (GBP) to United States dollars (USD) for your company InFusion America Inc.

Currency conversion rates were entered by an automatic load to the Daily Rates table. They can also be entered through a spreadsheet.

**Updating Currency Rates**

1. Navigate to the Period Close work area.
Use the Period Close work area to link to close processes and currency process.

2. Click the Manage Currency Rates link.
   Use the Currency Rates Manager page to create, edit, and review currency rate types, daily rates, and historical rates.

3. Click the Daily Rates tab.
   Use the Daily Rates tab to review and enter currency rates.

4. Click the From Currency list. Select the GBP - Pound Sterling list item.

5. Click the To Currency list. Select the USD - US Dollar list item.

6. Enter the dates for the daily rates that you are changing. Enter today’s date.

7. Click the Rate Type list. Select the Spot list item.

8. Click the Search button.

9. Click in the Rate field. Enter the new rate of 1.7 in the Rate field.

10. Click in the Inverse Rate field. Enter the new inverse rate of 0.58822 in the Inverse Rate field.

11. Click the Save button.

**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 ledgers, 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.

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:
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 implement Define Enterprise Structures for Human Capital Management, Project Management, and Sales Management.

**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

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.
Define Common Applications Configuration for Marketing

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 Generally Accepted Accounting Principles (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?

**Global Enterprise Structure Model**

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 Ltd.

- 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 and Customer Relationship Management 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**

**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.
To be able to use the Enterprise Structures Configurator, you must select the Enterprise Structures Guided Flow feature for your offerings on the Configure Offerings page in the Setup and Maintenance work area. 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**

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**
You can view a result of the interview process prior to loading the configuration. In the review results, you can view the divisions, legal entities, business units, reference data sets, and the management reporting structure that the application will create when you load the configuration.

**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.

**Rolling Back an Enterprise Structure Configuration: Explained**

The Enterprise Structures Configurator (ESC) provides the ability to roll back an enterprise configuration in the following circumstances:

**Manual Rollback**
You can manually roll back an enterprise configuration after loading it, for example, because you decide you do not want to use it. Clicking the Roll Back Configuration button on the Manage Enterprise Configuration page rolls back any enterprise structures that were created as a part of loading the configuration.

**Automatic Rollback**
If an error occurs during the process of loading the configuration, then the application automatically rolls back any enterprise structures that were created before the error was encountered.

**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. This figure illustrates the enterprise configuration that results from the analysis of InFusion Corporation.
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, expenses 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.
This table represents the selections that InFusion Corporation makes when specifying which legal entities to create on the Map Divisions by Country page.

<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 based 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>
</table>
| Country                                | • US
                                               • UK
                                               • Japan
                                               • India |
| Country and Division                    | • InFusion Lighting: Japan
                                               • InFusion Lighting: US
                                               • Infusion Security: UK
                                               • Infusion Security: India |
| 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.

**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>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>
</tbody>
</table>
Position Configuration for Marketing

Controlled (An industry that is highly structured in which all aspects of work and remuneration are well organized and regulated.)

<table>
<thead>
<tr>
<th>Positions</th>
<th>Positions</th>
<th>Positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturing</td>
<td>Positions</td>
<td>Jobs</td>
</tr>
<tr>
<td>Retail</td>
<td>Positions</td>
<td>Jobs</td>
</tr>
<tr>
<td>Education</td>
<td>Positions</td>
<td>Jobs</td>
</tr>
<tr>
<td>Other</td>
<td>Positions</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 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.

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.

**Uploading Locations Using a Spreadsheet**

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

**Synchronizing and Optimizing Database Search Indexes for CRM Objects: Explained**

You can synchronize and optimize database search indexes for your Oracle Fusion CRM objects to make your users' searches faster and more efficient. To do this, you can schedule two Enterprise Scheduler Service (ESS) jobs to perform the synchronization and optimization operations.

**Synchronizing Database Search Indexes**

The Synchronize Database Search Indexes for CRM Objects job is a common ESS job that refreshes text indexes for customer, contact, and party name, lead name, and opportunity name columns. It calls the CTX_SYNC_INDEX API to synchronize the text index to process inserts, updates, and deletes to the base table, and ensures that any changes made to these columns are reflected in the database index for the columns.
The recommended minimum interval for scheduling the database synchronization job is five minutes; you should allow a longer interval if possible.

You must have the CRM Application Administrator Duty role to see the Synchronize Database Search Indexes for CRM Objects job in the job list, and the Run Search Index Scheduler privilege to run the job.

To schedule the database synchronization job:

1. Navigate to the Scheduled Processes page by selecting Scheduled Processes under the Tools menu in the Navigator.

2. Click the Schedule New Process button.

3. Specify Job in the Type field on the Schedule New Process dialog box if it is not already specified.

4. Search for and select the job type of Synchronize database search indexes for CRM objects.
5. Click OK.

Optimizing Database Search Indexes

The Optimize CRM Search Indexes for CRM Objects job is a common ESS job that optimizes the search indexes. It calls the CTX_OPTIMIZE_INDEX API to perform the optimization, removing old data and minimizing index fragmentation, which can improve query response time.

You can run this job on a weekly, monthly, or as-needed basis if search performance is becoming poor. The recommended minimum interval for scheduling the database optimization job is five minutes; you should allow a longer interval if possible.

You must have the CRM Application Administrator Duty role to see the Optimize CRM Search Indexes for CRM Objects job in the job list, and the Run Search Index Scheduler privilege to run the job.

To schedule the database optimization job:

1. Navigate to the Scheduled Processes page by selecting Scheduled Processes under the Tools menu in the Navigator.

2. Click the Schedule New Processes button.
3. Specify Job in the Type field on the Schedule New Process dialog box if it is not already specified.

4. Search for and select the job type of Optimize database search indexes for CRM objects.

5. Click OK.

**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.
Define Common Applications Configuration for Marketing

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

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 during address entry suggested values will be provided for the address element, but the address element will not be 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 Error, then an address cannot 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.

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.

**Note**

The Geography Dimension value in territories is derived from sell-to addresses of sales accounts. To use geography dimensions in territories, ensure that the geography elements in addresses, such as state, city, and postal code, are validated. You can do so by enabling geography validation for each country using the Manage Geographies task. While doing so, ensure that at least one level in the geography hierarchy is enabled for geography validation. It is recommended that you enable geography validation for all geography levels that you intend to use for territory definition for each country. You can enable a list of values containing specific geography elements. This will help users search and select appropriate geography values during addresses entry and eliminate all possibilities of wrong address entry. You can also set geography validation control to Error in the Manage Geography Validation page. This ensures that users can only use valid geography elements in addresses. If you have already created addresses before setting up geography validation for a country, you must execute the Run Maintain Geography Name Referencing task for that country after enabling geography validation to ensure that all your geography elements are validated.
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.

Managing Geography Structures, Hierarchies, and Validation: Worked Example

This example shows how to configure the geography structure, hierarchy, and validation for a country geography, using the United Kingdom country geography as an illustration.

The following table summarizes the key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy an existing country structure?</td>
<td>No, create a new country structure.</td>
</tr>
<tr>
<td>What is the structure of the geography types?</td>
<td>Create geography types with the following ranking structure: 1. County 2. Post Town</td>
</tr>
<tr>
<td>What is the geography hierarchy?</td>
<td>Create the following hierarchy: 1. Country of United Kingdom 2. County of Berkshire 3. Post Town of Reading</td>
</tr>
<tr>
<td>Which address style format will you use when mapping geography validations?</td>
<td>The default address style format, called the No Styles Format.</td>
</tr>
<tr>
<td>Are you using Oracle Fusion Tax for tax purposes?</td>
<td>No, do not select Tax Validation for the geography types.</td>
</tr>
</tbody>
</table>

Add the County and Post Town geography types to the geography structure. Next, add the geographies for the County and Post Town geography types to define the geography hierarchy. Finally, specify the geography validations for the geography types you have added to the geography structure.
Defining the geography structure

Add the County and Post Town geography types to the United Kingdom geography structure.

1. On the Manage Geographies page, enter GB in the **Code** field. Click **Search**.
2. On the Manage Geographies page, click **Structure Defined**.
3. On the Manage Geography Structure page, click the **Create** button next to the **Copy Country Structure From** field.
4. In the Geography Structure section, select the County list item in the **Add Geography Type** field.
5. Click **Add**.
6. Select the Post Town list item in the **Add Geography Type** field.
7. Click **Add**.

Defining the geography hierarchy

To begin creating the geography hierarchy for the United Kingdom, you add the geographies for the County and Post Town geography types using the geography hierarchy user interfaces. You can also use the Manage File Import Activities task to import geography hierarchies using a csv or xml file.

1. On the Manage Geographies page, enter GB in the **Code** field. Click **Search**.
2. On the Manage Geographies page, click **Hierarchy Defined**.
3. On the Manage Geography Hierarchy page, Geography Hierarchy section, click the United Kingdom to highlight the table row.
4. Click the **Create** button.
5. In the **Create County** page, Primary and Alternate Names section, enter Berkshire in the **Name** field.
   6. Click **Save and Close**.
7. On the Manage Geography Hierarchy page, Geography Hierarchy section, click Berkshire to highlight the table row.
8. Click the **Create** button.
9. In the **Create Post Town** page, Primary and Alternate Names section, enter Reading in the **Name** field.
   10. Click Save and Close.

Defining the geography validations

Now you want to specify the geography validations for the geography types you have added to the United Kingdom. Define the geography mapping and validation for the United Kingdom default address style format. Then map the geography types to attributes, enable the geography types for Lists of Values and Geography validation, and set the geography validation level.

1. On the Manage Geographies page, click **Validation Defined**.
2. On the Manage Geography Validation page, Address Style section, click No Styles Format to highlight the table row.

3. For the County geography type, click the County list item in the Map to Attribute field.

4. Click the Enable List of Values option for the County geography type.

5. Click the Geography Validation option for the County geography type.

6. For the Post Town geography type, click the City list item in the Map to Attribute field.

7. Click the Geography Validation option for the Post Town geography type.

8. In the Geography Validation Control section, click the Error list item in the Geography Validation Level for Country field.

9. Click Save and Close.

**FAQs for Define Geographies**

**When do I define address cleansing?**

When address data entered into the application needs to conform to a particular format, in order to achieve consistency in the representation of addresses. For example, making sure that the incoming data is stored following the correct postal address format.

**Why can’t I update a geography structure by copying an existing country structure?**

You can only update a geography structure by adding existing geography types, or by creating new geography types and then adding them to the geography structure. You can only copy an existing country structure when you are defining a new country structure.

**Why can’t I delete a level of the country geography structure?**

If a geography exists for a country geography structure level then you cannot delete the level. For example, if a state geography has been created for the United States country geography structure, then the State level cannot be deleted in the country geography structure.

**Can I add any geography to the geography hierarchy?**

Yes. However, the geography type for the geography that you want to add must be already added to the country geography structure.
Can I edit a specific geography in the geography hierarchy?

Yes. In the Manage Geography Hierarchy page you can edit details such as the geography’s date range, primary and alternate names and codes, and parent geographies.

How can I add a geography that is the level below another geography in a geography hierarchy?

Select the geography that you want your geography to be created below, and then click the Create icon. This will allow you to create a geography for a geography type that is the level below the geography type you selected. The structure of the country’s geography types are defined in the Manage Geography Structure page.

Define File-Based Data Import

File-Based Data Import: Highlights

You can import application data from external sources into the Oracle Sales Cloud database by using the Define File-Based Data Import group of tasks available from the Setup and Maintenance work area. The two primary sources of documentation on file-based data import are:

- My Oracle Support Article 1564536.1 (Oracle Sales Cloud File-Based Data Import Guide) at https://support.oracle.com
- Oracle Enterprise Repository assets of type File Based Data Import at http://fusionappsoer.oracle.com

Oracle Sales Cloud File-Based Data Import Guide

The Oracle Sales Cloud File-Based Data Import Guide is a collection of topics that aids you in using file-based data import by providing:

- High-level information about the file-based data import process, architecture, and tools
- Detailed information for importing specific objects and their related objects
- Guidance on importing a minimal set of fields for specific objects
- References to Oracle Enterprise Repository assets that support import of specific objects

Oracle Enterprise Repository Assets

Oracle Enterprise Repository assets of type File Based Data Import provide:
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.

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, expenses 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.
One Payroll Statutory Unit with Several Tax Reporting Units That Are Independent from the Legal Employer

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

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 based 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.

**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 Oracle 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.

The following business functions generate financial transactions and will require a primary ledger and a default legal entity:

- Billing and revenue management
- Collections management
- Customer payments
- Expense management
- Materials management
- Payables invoicing
- Project accounting
- Receiving
- Requisitioning

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.

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.

Uploading Locations Using a Spreadsheet
If you have a list of locations already defined for your enterprise, you can upload them from a spreadsheet. To use this option, you first download a spreadsheet template, add your location information to the spreadsheet, and then upload directly to your enterprise configuration. You can upload the spreadsheet multiple times to accommodate revisions.

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.

Adding a New Division After Acquiring a Company: Example

This example shows how to restructure your enterprise after acquiring a new division.

Scenario

You are part of a senior management team at InFusion Corporation. InFusion is a global company with organizations in the United States (US), the United Kingdom (UK), France, China, Saudi Arabia, and the United Arab Emirates (UAE). Its main area of business is in the high tech industry, and it has just acquired a new company. You must analyze their current enterprise structure and determine what new organizations you need to create to accommodate the new company.

Details of the Acquired Company

The acquired company is a financial services business based in Germany. Because the financial services business differs significantly from the high tech business, you want to keep the financial services company as a separate business with all the costs and reporting rolling up to the financial services division.

Analysis

The following table summarizes the key decisions that you must consider when determining what new organizations to set up and how to structure the enterprise.

<table>
<thead>
<tr>
<th>Decision to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create location?</td>
<td>The financial services company is based in Frankfurt as are the departments, so you need to create only one location.</td>
</tr>
<tr>
<td>Question</td>
<td>Answer</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create separate division?</td>
<td>Yes. Although the new division will exist within the current enterprise structure, you want to keep the financial services company as a separate line of business. Creating a separate division means you can manage the costs and reporting separately from the InFusion Corporation. It also means you do not have to modify any existing organizations in the enterprise setup.</td>
</tr>
<tr>
<td>Create business unit?</td>
<td>Yes. The financial services business requires you to create several jobs that do not exist in your high tech business. You can segregate the jobs that are specific to financial services in a new business unit.</td>
</tr>
<tr>
<td>How many departments?</td>
<td>The financial services company currently has three departments for sales, accounting, and marketing. As you have no plans to downsize or change the company, you can create three departments to reflect this structure.</td>
</tr>
<tr>
<td>How many cost centers?</td>
<td>Although you can have more than one cost center tracking the costs of a department, you decide to create one cost center for each department to track costs.</td>
</tr>
</tbody>
</table>
| How many legal entities?                     | 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. In this case, you need only one legal entity.  
You must define the legal entity as a legal employer and payroll statutory unit. As the new division operates in Germany only, you can configure the legal entity to suit Germany legal and statutory requirements. |
| Create legislative data group?               | Yes. Because you currently do not employ or pay people in Germany, you must create one legislative data group to run payroll for the workers in Germany. |

**Resulting InFusion Enterprise Structure**

Based on the analysis, you must create the following:

- One new division
- One new location
- Three new departments
- Three new cost centers
- One new legal entity
One new legislative data group

The following figure illustrates the structure of InFusion Corporation after adding the new division and the other organizations.

---

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: Explained

As part of your initial implementation, you specify whether to use jobs and positions, or only jobs. Jobs are typically used without positions by service industries where flexibility and organizational change are key features.

Basic Details

Basic details for a job include an effective start date, a job set, a name, and a code. A job code must be unique within a set. Therefore, you can create a job with the code DEV01 in the US set and another job with the same code in the UK set. However, if you create a job with the code DEV01 in the Common set, then you cannot create a job with the same code in any other set.

Benchmark Information

You can identify a job as being a benchmark job. A benchmark job represents other jobs in reports and salary surveys. You can also select the benchmark for jobs. Benchmark details are for informational purposes only. A progression job is the next job in a career ladder.

Progression Information

Progression jobs enable you to create a hierarchy of jobs and are used to provide the list of values for the Job field in the Promote Worker and Transfer Worker tasks. The list of values includes the next three jobs in the progression job hierarchy. For example, assume that you create a job called Junior Developer and select Developer as the progression job. In the Developer job, you select Senior Developer as the progression job. When you promote a junior developer, the list of values for the new job will include Developer and Senior Developer. You can select one of these values, or select another one.

Jobs and Grades

You can assign grades that are valid for each job. If you are using positions, then the grades that you specify for the job become the default grades for the position.

Evaluation Criteria

You can define evaluation criteria for a job, including the evaluation system, a date, and the unit of measure for the system. One predefined evaluation system is available, and that is the Hay system. An additional value of Custom is included in the list of values for the Evaluation System field, but you must add your own criteria and values for this system.
Uploading Jobs Using a Spreadsheet

If you have a list of jobs 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 job information to the spreadsheet, and then upload directly to your enterprise configuration. You can upload the spreadsheet multiple times to accommodate revisions.

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

Schedule Components: How They Fit Together

Schedules are comprised of workday patterns and exceptions. Workday patterns are comprised of shifts. You can also create exceptions, nonworking days, to the schedules.

Begin by creating shifts and then assigning those shifts to workday patterns. Next, create a schedule that is a collection of workday patterns and any exception dates.

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 a.m. to 11 a.m. 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 the change affects a schedule, that is they are associated to a schedule. If a shift is created but not assigned to a pattern (or assigned to a pattern but the 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.

**Managing Duration Workday Patterns**

A new group of employees starts 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

Getting Started with Oracle Fusion CRM Collaboration: Overview

Oracle Fusion CRM Collaboration is a collection of features of the Oracle WebCenter product. Employees in an organization can use these features to collaborate and stay informed of the latest information.

Employees can use Oracle Fusion CRM Collaboration features to do the following business tasks:

• Access real-time feeds of business transactions.
• Engage in contextual discussions around key business goals.
• Leverage group spaces to enhance team collaboration.
• Enhance virtual team communication by staying up-to-date on peer activities, deals, connections, and community updates.
• Gain greater visibility into opportunities and best practices.
• Make informed decisions with peer feedback.
• Access critical information to streamline the sales process.

From Oracle WebCenter, Oracle Fusion CRM Collaboration leverages the following features.

Activity Streams

Activity streams provide an ongoing view of activities from connections, actions taken in group spaces, and other business activities. Within Oracle Fusion CRM Collaboration, activity streams have been enabled for opportunities and customer business events, as displayed in the following Oracle Fusion CRM areas:

• Welcome page
• Sales Dashboard
• Customer Center overview page
Employees can use activity streams to:

- Stay current on relevant business updates or updates from the communities of which the user is a member.
- Stay up-to-date on what others are working on and what is happening in your organization.
- Make timely decisions with real-time contextual insight.
- Overcome organizational and geographical boundaries that make it hard to find help and expertise.
- Increase employee interaction and collaboration.

For more information, see "Tracking Your Connections Activities" in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces.

**Blogs**

Blogs are typically personal records of an individual user's experience and opinions. In Oracle WebCenter Spaces, users can create blogs to group related blog posts, for example to group topics by the same author or otherwise related topics. Each blog contains various blog posts. Although, blogs have not been exposed in Oracle Fusion CRM, users can access blogs by way of Oracle WebCenter Spaces. For more information, see "Working with Blogs" in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces.

**Discussion Forums**

You can use a discussion forum to post, respond to, and preserve topical information, and other users can post information relevant to those topics. All of this information is preserved within the discussion forum so that teams of users can collaborate on opportunities and customer records. Everyone who is part of the opportunity or customer team can post and reply to discussions. Within Oracle Fusion CRM Collaboration, discussions are enabled for opportunities and customer business objects. For more information, see "Creating a Discussion Forum" in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces.

Discussions has been enabled for:

- Opportunity
- Customer
- References
- Competitors

**Group Spaces**

Teams of employees can use group spaces to collaborate in the context of an opportunity or customer record. Within a group space, employees can collaborate by engaging in existing discussions or creating new ones, and they can create new wikis. They can also create new group spaces and leverage available features, such as discussions, blogs, wikis, announcements, and so on. By using group spaces, employee groups can contend with work blockages, such as:
- Unstructured sales activities that are not captured.
- Information that is locked in silos.
- Hard-to-find successful or useful content.
- Numerous e-mail threads used for collective content authoring.

For more information, see "Understanding Space Basics" in the Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces.

**Tags**

Tags categorize business objects in Oracle Fusion Applications, for example specific invoices and opportunities, with your own keywords. You can share tags so that anyone searching or browsing for items can find them based on common tags. For more information, see "Working with the Tags Service" in the Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces.

**Wikis**

Wikis are Web sites of interlinked Web pages that members of a group can create and edit in a Web browser. Members who have the appropriate permissions can add and edit information to share it with their group. For more information, see "Working with Wikis" in the Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces.

Wikis have been enabled for:

- References
- Competitors

**Oracle Fusion CRM Social Network Administration: Overview**

Oracle Social Network allows you to create Social Objects and Conversations that are associated with an Oracle Fusion CRM object record. Social Objects are records within a business application or business process that are mapped into Oracle Social Network. You can expose a record from an Oracle Fusion CRM application in Oracle Social Network and make it visible to selected people.

Within Oracle Fusion CRM, you can have a Conversation on the same page where the record appears. Conversations are online discussions that can include messages, replies, documents, links, and gadgets. They provide the central point of collaboration in Oracle Social Network, bringing people together to discuss, evolve, and preserve all of the exchanges leading to discussions, plans, and partnerships. You can associate Conversations (called Related Conversations) to a Social Object, making them easy to find.

You can integrate Oracle Fusion CRM with Oracle Social Network by enabling end users to create Oracle Social Network Social Objects based on object records in Oracle Fusion CRM. By default, Oracle Fusion CRM objects are not enabled for Oracle Social Network. This topic explains how to enable objects so that users can create Oracle Social Network Social Objects based on object records, and view object-related status updates.
The following table shows the Oracle Fusion CRM objects you can enable for Oracle Social Network:

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Social Object Group Name</th>
<th>Social Object Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Trading Community Organization</td>
<td>Trading Community Organization Profile</td>
</tr>
<tr>
<td>Person</td>
<td>Trading Community Person Information</td>
<td>Trading Community Person Information Profile</td>
</tr>
<tr>
<td>Group</td>
<td>Trading Community Group</td>
<td>Trading Community Group Profile</td>
</tr>
<tr>
<td>Competitor</td>
<td>Sales Competitor Service</td>
<td>Sales Competitor</td>
</tr>
<tr>
<td>Lead</td>
<td>LeadAM</td>
<td>Sales Lead Version 2</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Service</td>
<td>Opportunity</td>
</tr>
<tr>
<td>Partner Account</td>
<td>PartnerCenterPartnerProfileAM</td>
<td>Partner</td>
</tr>
<tr>
<td>Reference Customer</td>
<td>Sales Reference Customer</td>
<td>Sales Reference Customer</td>
</tr>
<tr>
<td>Sales Account</td>
<td>Sales Party</td>
<td>Sales Account</td>
</tr>
</tbody>
</table>

You can enable entire objects or only certain attributes. If you only enable a subset of an object’s attributes, only the enabled attributes will display their status change information in the Oracle Social Network feed.

When an object is enabled, you can set it to either Automatic or Manual sharing.

- Automatic sharing means that the Social Object in Oracle Social Network that corresponds to the Oracle Fusion business object is created automatically when the object record is created. If the object record already exists before the Oracle Fusion object is enabled for Oracle Social Network, then the next update creates the object in Oracle Social Network. Otherwise, users will see a Share button that they can click to enable the object record for Oracle Social Network. If an object record has already been shared and the user has permission, a Join button is displayed. Clicking this button allows the user to be added as a member of the existing Social Object associated with that object record.

- If an object is set to Manual sharing, the Oracle Fusion business object is not created automatically when the object record is created. Users can manually decide whether they want to share the business object in Oracle Social Network by clicking a Share button on the object record. For example, if the Opportunity object were set to Manual sharing, users would have to manually click the Share button on an individual opportunity object record to expose it to Oracle Social Network. If an object record has already been shared and the user has permission, a Join button is displayed. Clicking this button allows the user to join (be added as a member of) the existing Social Object associated with that object record.

All users have access to Oracle Social Network; however, only users with the CRM Application Administrator job role can configure the set of objects and fields that are enabled for Oracle Social Network sharing. By default, the
Application Administrator job role is configured with the Oracle Social Network Management duty role.

To enable an Oracle Fusion object for Conversations:

1. Navigate to Administration, Setup and Maintenance.
2. Access the Manage Oracle Social Network Objects task.
3. Click the Manage Oracle Social Network Objects tab.
4. Select the object you want to enable.
5. Click the Enable Object button.
6. Click Save.

A dialog box displays where you can specify whether to set the object to Automatic or Manual sharing, or to turn off sharing entirely.

You can also select individual attributes on an object to include in Social Objects. For example, you might want to display when a lead’s status changes, but not the name of the person who made the change.

To enable individual attributes:

1. After you have selected an object on the Manage Social Network Objects tab, expand the Attributes section for the object.
2. Click the New icon.
3. Choose the attributes you want to include by checking their Enabled check boxes.
4. Click Save.

You must enable at least one attribute; by default, all attributes are disabled regardless of which option you have selected for sharing. For verification, a green check mark appears next to object names that have been successfully enabled. If an object is not successfully enabled, a yellow warning icon appears next to it.

To disable Oracle Social Network for an enabled object, select the object and click Disable Object.

You can enable all available objects by clicking the Enable All button. However, note that this option is not limited to Oracle Fusion CRM objects, but will enable Oracle Social Network sharing for objects from all other eligible Oracle Fusion applications that you have licensed.

**Collaboration Capabilities in Oracle Fusion CRM: Highlights**

Oracle Fusion CRM Collaboration includes the following features.

**Fusion Welcome Page**

Across all Fusion applications, the **Fusion Welcome** page shows all activity stream messages:

- Opportunities and customer records that a user is following.
- Discussion groups of which the user is a member.
• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available in a group space depend on the template that was used when the group space was created.

• Connections updates.

• Status updates.

• Healthcare management activity stream messages.

• Projects.

• All corresponding comments and likes for any of the above activity messages.

**Sales Dashboard Page**

The Sales Dashboard shows activity stream messages for the following functions:

• Opportunities and customer records that a user is following.

• Discussion groups of which the user is a member.

• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available on a group space depend on the template that was used when the group space was created.

• Connections updates.

• Status updates.

• All corresponding comments and likes for any of the above activity messages.

**Customer Center Overview Page**

The Customer Center Dashboard shows activity stream messages for the following functions:

• Opportunities and customer records that a user is following.

• Discussion groups of which the user is a member.

• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available on a group space depend on the template used when the group space was created.

• Connections updates.

• Status updates.

• All corresponding comments and Likes for any of the above activity messages.

**Posting Messages**

You can post a status update message from the Oracle Fusion Welcome page. This message can be posted to Everyone, to members of a specific group space, or to members of a specific opportunity or customer sales team. A user can also post a message from an Oracle WebCenter group space. Within a group space, the same field is available for users to post a status update.

• For more information on Oracle Fusion CRM Collaboration capabilities, See: Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces
Group Spaces in Oracle Fusion CRM: Highlights

Group spaces bring people together in a virtual environment for ongoing interaction and information sharing, enabling the formation and support of social networks. Teams can use group spaces to collaborate through discussions, in the context of an opportunity or customer record, and create new wikis. Users can also create a new group space and leverage available features, such as discussions, blogs, wikis, announcements, and so on. Group spaces support the collaboration of project teams and communities of interest by providing a dedicated and readily accessible area for relevant services, pages, and content, and by supporting the inclusion of specified members. A member can invite you to join a space, in which case you receive a notification. To see all the spaces that you can join, use the Spaces link in the global area or go to the Spaces dashboard, which contains a list of the spaces to which you belong, a list of any recommended spaces, and a tool to search for all public spaces that you can join.

The available features in a group space depend on the template that was selected when the group space was created. Out-of-the-box templates provide a variety of features. Spaces are an Oracle WebCenter feature that is used not only in the Oracle WebCenter Spaces application but also throughout Oracle Fusion Applications.

Accessing Group Spaces

You can access group spaces by the Spaces link in the global navigation area. From there, you can access the existing public group spaces or the group space of which you are a member. From within the Edit Opportunity page of an opportunity record, you can also access an existing group space or create a new group space. Click an icon to display a popup window where you can do the following group space activities:

- Select and add a group space: You can select an existing group space from the list. The application allows all users who have access to the opportunity record to use the features exposed in the selected group spaces.
- Create and link a group space: You can create new group spaces using existing group spaces templates. Users who have access to the opportunity record can use the features that are exposed in the newly created group spaces.

Setting Up Group Spaces

You can associate existing group spaces to an object, or you can create new ones. These associations have no impact on any embedded group spaces within the object, nor do they create a new embedded forum or wiki for that object. You can make associations in the following ways:

- Opportunity management: A default group space is associated with each opportunity record. Drill into the group space to use any of the features available in the default group space (wikis and discussions). You can create and associate new group spaces to the opportunity record. Depending on the template that was selected when the group space was
created, you can use any of the features in the group space, such as wikis, discussions, blogs, announcements, and so on.

- Group Spaces link: From the top navigation menu, you can create and access a group space and use any of the features available in the group space.

Security for Enabling or Viewing Group Space Widgets

Users need specific security permissions to enable or view group space widgets, such as forums, wikis, and so on, for the following objects:

- Opportunity: For enabling, the typical role is a sales person or other advanced user. Security is controlled by an opportunity team member with full access plus the functional privilege View Opportunities. For viewing, the typical role is a sales person with regular access. Security is controlled by an opportunity team member with View or Edit access plus the functional privilege View Opportunities.

- Competitor: For enabling, the typical role is a sales administrator. Security is controlled with the functional privilege Manage Sales Competitor. For viewing, the typical role is a sales person with the functional privilege View Sales Reference.

- Reference: For enabling, the typical role is sales administrator. Security is controlled with the functional privilege Manage Sales Reference. For viewing, the typical role is a sales person with the functional privilege View Sales Reference.

Optional Enabling

Deploying organizations have a choice of enabling embedded forums and wikis for sales objects by using two profile options:

- Key: (<Object> Group Space Enabled) Determines whether or not a particular object, such as an opportunity, has support for embedding forums and wikis.

- Key: (<Object> Group Space Template Default) Determines which group space template to use in case the object is to have embed support (if the value for the first option is Y.)

- For an embed-enabled sales object, the edit view for an instance of that object provides an additional check box to show existing embedded forums and wikis (or enable embedding and showing forums and wikis if no embedded widgets exist at that point). Only advanced users of the object instance (with full access for opportunities, sales administrators for competitors and references) can see and interact with this check box. It is not visible to other users.

Embedding Group Space Widgets

To embed group space widgets, use the following actions:

- When embedded forums or wikis already exist, display the Discussions tab and Wikis tab with existing embedded content by checking the Discussions and Wikis check box. Forums and wikis are created for that object instance using the group spaces template that was specified in the first profile option. The Discussions tab and Wikis tab appear in the object instance for the newly-created widgets.
• To hide The Discussions tab and Wikis tab with existing embedded content, uncheck the Discussion and Wikis check box.

**Group Spaces and Oracle WebCenter**

For more information on group spaces, see the Oracle Fusion Middleware User's Guide for Oracle WebCenter. As you read that guide, note the following points:

• Disregard content that is specific to the Oracle WebCenter Spaces application, for example the home space. Although there is no home space in Oracle Fusion Applications, many of the features within the home space are also available in Oracle Fusion Applications.

• The way to access spaces in Oracle Fusion Applications is different from the instructions in the guide. In Oracle Fusion Applications, you can use the Spaces link in the global area or on pages where available, as well as the Spaces dashboard.

See: Oracle Fusion Middleware User’s Guide for Oracle WebCenter

**Discussion Forums in Oracle Fusion CRM: Highlights**

Teams of users can use discussion forums to collaborate on opportunities and customer records. Everyone who is part of an opportunity or customer team can collaborate and take part in discussions by posting and replying to discussions. When someone posts a new discussion entry or replies to an existing discussion, the users who are members of the opportunity or customer team are notified by an activity stream message.

Members can drill into a discussion to read all replies, watch the forum where discussions are created, and view, edit, reply, and create their own discussions by accessing the group space. Members can also create and participate in text-based discussions with other users within the scope of specific business objects, an application, or a space. For example, in Oracle Fusion Applications Help, you can post questions or comments regarding topics covered by a specific help file and view posts from other members regarding the same help.

The Discussions feature is an Oracle WebCenter service used within and outside of spaces in Oracle Fusion Applications. Discussions are fully described in the Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces. As you read that guide, note the following points:

• In Oracle Fusion Applications, discussions can exist outside of spaces. Content that is specific to spaces applies to spaces within Oracle Fusion Applications, but discussions work similarly outside of spaces.

• Various user interfaces, or task flows, are described in the guide. The interfaces and flows that are available to you depend on the pages that you can access.

**Setting Up and Using Discussion Forums**

Moderators can create discussion forums in which members can view and participate in topics and threads. Set up and use discussion forums in the following ways:
• Edit Opportunity page: This option is available only to users at the level of someone like an opportunity owner, that is, a member with full access. To enable a discussion forum for an opportunity, click an Opportunity Name. Then in the Edit Opportunity page Summary region, check the Show Discussions and Wiki check box and then Save. This option creates discussions and wikis for the opportunity, and a Discussions tab in the Additional Details region of the Edit Opportunity page. After you enable discussions, you cannot disable them. When a discussion is enabled on the opportunity, anyone with a minimum of view access on the opportunity can contribute to it.

• Customer Center: For each customer record, the Discussion Forums link is displayed in the Customer Center region. A setup is not necessary.

• Opportunity Management: In each opportunity record the Discussions tab is available in the Additional Details region. You can create and associate new group spaces to the opportunity record and the user Discussions feature under the group space.

• Global group space: At the global navigation level, click the Spaces link, then Create. When you create a space choose the Discussion template, then you can navigate to the space, add members, create pages, and upload documents.

See Also
• For more information about discussion groups,

See: Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces

Wikis in Oracle Fusion CRM: Highlights

A wiki is a Web site of interlinked Web pages that allows members of a group to create and edit the Web pages in a Web browser. Members of a group who have the appropriate permissions can add and edit information and share it among the group. Wikis are available by way of CRM References and Competitors.

In Oracle WebCenter Spaces, you can create and manage wiki documents by using the Documents service. To support the wiki functionality, the Documents service stores all wiki documents on Oracle Content Server. For this feature to work, the Oracle Content Server needs to be set up.

Setting Up Wikis

This option is available only to users at the level of full access, for example, someone like an opportunity owner.

• In the Edit Opportunity page, to enable a wiki for an opportunity, click an Opportunity Name.

• Then in the Edit Opportunity page Summary region, check the Show Discussions and Wiki check box and then save.

• This option creates wikis for the opportunity and a Wiki tab about competitors. When a wiki is enabled on the opportunity, anyone with a minimum of view access on the opportunity can contribute to it.
• After you enable wikis, you cannot disable them.

See Also
• For more information about wikis, see the following document.
  See: Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces

Tags in Oracle Fusion CRM: Highlights

Use tags to categorize business objects in Oracle Fusion Applications, for example specific invoices and opportunities, with your own keywords. You can share tags so that anyone searching or browsing for items can find them based on common tags. For example, members collaborating on a project can tag all related work with a particular term. Although tags are available to anyone who has access to the item, when you create tags you can designate them to be private.

The tags feature is an Oracle WebCenter service.

Tags are fully described in the Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces. As you read that guide, note the following points:

• Disregard discussions specific to the Oracle WebCenter Spaces application, for example the global search in the application.

• Although the guide describes tagging Oracle WebCenter pages and documents, in Oracle Fusion Applications you tag specific business objects.

• Not all tag features are available in Oracle Fusion Applications, for example the Tags and Similarly Tagged Items user interfaces.

Understanding Tags

Aside from the Tags icon and the Tag Center dialog box, tags are available in Oracle Fusion Applications from applicable Oracle Fusion Applications Search results. The search considers tags in finding matching results. The search results do not indicate the number of times a particular tag was applied, and not all searches retrieve private tags. You can access the Tag Center dialog box from the global area or other places in Oracle Fusion Applications, for example in Oracle Fusion Applications Help. The Tag Center shows all tags, not only those relevant to what you are working on. For example, the tag center in Oracle Fusion Applications Help displays all tags, not only the tags for help files. Use tags to do the following actions:

• Use the tag icon to tag specific business objects. Read only about the step that describes the fields you enter to tag an item.

• Browse and search for items using tags in the Tag Center dialog box. Disregard instructions about how to open the tag center.

• For more information about tags,
  See: “Understanding the Tag Center” in Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces
  See: "Working with Tags and Tagged Items in the Tag Center” in Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces
See: "Tagging Application Pages" in Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces

**Activity Streams in Oracle Fusion CRM: Highlights**

Activity streams show a continuous view of activities from connections, actions in group spaces, and other business activities. For example, you might see that specific users have created, edited, or deleted specific business objects, such as customers or opportunities, and you can use activity streams to access more information about that user or object. Additionally, similar to social networking Web sites, the activity stream displays messages that other users want to broadcast.

Activity Stream is an Oracle WebCenter feature, available in Oracle Fusion Applications on the **Welcome** dashboard and various other locations within and without the context of spaces. The types of activities that are tracked vary depending on each Activity Stream region.

**Setting Up Activity Streams**

You can set up activity streams from the **Navigator** link by selecting **Setup and Maintenance** under **Tools**. In the **Search: Tasks** field, type "Activity". In the **Search Results** list select **Set Activity Streams Options**. Click the icon in the **Go to Task** column to open the **Setup and Maintenance** page. Expand **Service Categories**. In the rows for **Customer** and **Opportunity** select the following check boxes:

- **Display Activities**: When new updates are made, the application displays the corresponding activity messages on the application UI.
- **Allow Owner Override**: This function allows the end user to override the Display Activities option by going to the Activity Streams Options feature.
- **Publish**: The system publishes the activity stream messages when new updates are made. These messages are first published or written to the Activity Streams repository, where they are written to the database table but not displayed. To display them, click Display Activities.

**Understanding Activity Stream Security**

Activity stream messages are only displayed to users who are:

- Following the opportunity or customer record (specific to CRM).
- Members of a discussion group who are involved with a discussion entry that is associated to an opportunity or customer record. In this case, users who have been invited to be part of the discussion receive an activity stream message for create, read, update, and delete operations about the discussion. The events tracked for opportunities are different from the ones tracked for customers.
- Connected to specific users.

• Members of a group space.

Tracking Activity Streams

Users can start receiving activity streams for opportunities or customer records by accessing the Opportunity or Customer Edit page. Under the header, click Follow. The system starts displaying the activity stream for changes to the records that the user is following. When using activity streams, note the following points:

• A user can stop receiving activity stream messages for opportunities or customer records by unfollowing the opportunity or customer record. Hover over the message and click Unfollow.

• You can hide (not display) updates to connections by hiding the person. You can also hide the activity stream messages from the group spaces that are associated to the opportunity or customer record.

• To view or unhide the connection or group space, click the Options link to see a list of connections or group spaces that are hidden. If you choose to unhide them, the system displays the connections or group space activity stream messages again.

• When an activity stream message is displayed, the message contains hyperlinks to the user who updated the record. Clicking that user's name returns the user's Profile page. If the activity stream is for an opportunity record, then the hyperlink is to that opportunity record. If you click an opportunity name, then the application returns the Opportunity Edit page. If the activity stream is for a customer record, then the hyperlink is to that customer record. If the you click a customer name, then the application opens the Customer Edit page.

Liking and Commenting in an Activity Stream

• For information on liking and commenting in an activity stream,


Activity Stream and Oracle WebCenter Spaces

Activity streams are described fully in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces. As you read that guide, note the following points:

• Although the guide describes activity streams only in the context of spaces, in Oracle Fusion Applications activity streams work similarly outside of spaces.

• The scope of what is potentially tracked in activity streams is different from what is described in the guide. Many types of activities are specific to Oracle Fusion Applications.

• The exact navigation or user interface in Oracle Fusion Applications might differ from what is described in the guide. Ignore any subject matter that is specific to Oracle WebCenter Spaces.

Using Profiles

In an activity stream message, click a person’s name or picture to open the Portrait tab giving general information about that person, such as contact information and areas of expertise. If you click on your own name or picture, then the My Portrait tab opens, showing more detailed, human resources-related information about you, such as your professional development goals.

- For more information about profiles,

Blogs in Oracle Fusion CRM: Highlights

Blogs are typically personal records of an individual user’s experience and opinions. The word "blog" is a contraction of the term "Web log". In Oracle WebCenter Spaces, you can create blogs to group related blog posts, for example to group topics by the same author or related subjects.

In Oracle WebCenter Spaces, each blog contains various blog posts, with the most recently added blog post displayed at the top. In a Blog page style mode, the Archives pane on the right provides controls for navigating to blog posts. Blog posts are categorized automatically by year and subcategorized by month. Next to the date of a year, the number in parenthesis represents the total number of blog posts created during that year. The number next to a month represents the total number of blog posts created during that month. For example, the designation December (2) shows that two blog posts were created in the month of December. Clicking any month displays all blog posts that were created during that month. You can view all blog posts available in a blog, regardless of the month or year they were created, by clicking the View All link. If there are more blog posts than can be displayed on one page, typically 10 posts, you can use the Next and Previous buttons to navigate through the list of blog posts.

See Also
- For more information on blogs, see the following document.
  See: Oracle Fusion Middleware User’s Guide for Oracle WebCenter Spaces

Removing the Oracle Social Network Conversations Tab and Tree Node in Oracle Fusion CRM: Worked Example

Oracle Social Network Conversations are online discussions that can include message, replies, documents, links, and gadgets. They can be shared among people who have permission to view them and who have chosen to join them. In the current release, if an Oracle Fusion CRM object is enabled for Oracle Social Network sharing, then the Oracle Social Network UI is visible on the object record. If you previously used the Oracle Fusion CRM Application Composer to manually add the Oracle Social Network UI to Customer Center and Sales, you
must now use Application Composer to remove it. This is because in this release, it is automatically added after your administrator has enabled the Oracle Fusion Application object for Oracle Social Network sharing during the setup process.

This topic explains how to remove the Oracle Social Network UI that was manually added to Oracle Fusion Customer Center and Oracle Fusion Sales using Oracle Fusion CRM Application Composer.

**Note**

Never make your customizations directly in the mainline. Instead, always use sandboxes whenever possible.

**Removing the Conversations Subtab for Opportunity**

1. Sign in to Oracle Fusion CRM as an administrative user. You must have the Master Data Management Application Administrator job role and duty role.

2. From the Navigator, navigate to Opportunities.


4. On the Opportunity Edit page, select Customize Opportunities Pages from the Administration menu.

5. Select Site and click OK.
6. Choose Select mode.

7. Expand the Additional Details section if it is not already expanded.

8. Click the blank area to the right of the Conversations subtab.


10. Select and highlight Conversations in the Reorder the tabs choice list.
11. Check the Hide this tab check box to hide the subtab for all users.

12. Click OK.

**Removing the Conversations Tree Node for Sales Account**

1. Sign in to Oracle Fusion CRM as an administrative user. You must have the Master Data Management Application Administrator job role and duty role.

2. Select Customers from the Navigator menu.
3. Select any Sales Account record.

4. On the Customer Edit page, select Customize Customers Pages from the Administration menu.

5. Select the Site level and click OK.

6. Choose Design mode.
7. Click the Action button and select Manage Customer Tree.

8. Highlight Conversations in the Tree Nodes list.

9. Deselect the Visible check box under Conversations: Details to hide the node for all visitors.
10. Click Save.

FAQs for 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 Security for Customer Relationship Management

**Oracle Fusion CRM Security: Functional Overview**

Oracle Fusion CRM Applications come secured using the industry standard for access control that is called role-based access control (RBAC). This topic discusses key aspects of the RBAC approach that are specific to an Oracle Fusion CRM implementation. You must review other documentation to understand how RBAC is designed to handle a broad range of security needs.
The RBAC standard supports the enforcement of user access control that is based on the role of the user within the organization rather than the user’s individual identity. In RBAC, you assign users with roles that represent the job functions in your enterprise. These roles provide access both to the application functions that users need to perform their jobs as well as the permissions to access the data where they need to perform those functions.

Oracle Fusion Applications, including Oracle Fusion CRM, are secured with a predefined set of enterprise roles. This security reference implementation fulfills the needs of midsize horizontal enterprises, generally between 250 and 10,000 employees. To enable users to perform specific jobs in your CRM enterprise, you provision them with the appropriate enterprise roles. For example, when you provision sales managers with the sales manager job role, they can perform all their job duties, including managing sales teams and their forecasts, setting quotas, and managing sales leads and opportunities.

You can change this security reference implementation if the roles in your enterprise are different or if you want to accommodate expansion into vertical industries, such as health care, insurance, automobiles, or food manufacturing. Application patching does not affect your changes.

Key Components of Oracle Fusion CRM Application Security

The following graphic provides an overview of the key components that determine what functions users can perform in the application (functional security) and on what data they can perform those functions (data security).

For Oracle Fusion CRM, the relevant security components are the following:
• User

This document refers to security for human users, not background processes and other system users that are also secured with RBAC.

• Autoprovisioning rules

Rules automatically provision users with enterprise roles that carry all the security settings that are appropriate for their duties. You can provision users based on their role in your organization and other factors, such as their status as an employee or contractor.

• Enterprise roles

There are two types of enterprise roles:

• Job roles

Job roles permit users to perform activities specific to their job. For example, providing users with the Sales Manager job role permits them to manage salespersons within the organization, follow up on leads, generate revenue within a territory, build a pipeline, manage territory forecasts, and assist salespersons in closing deals.

• Abstract roles

Abstract roles permit users to perform functions that span the different jobs in the enterprise. For example, each user who is an employee must be provisioned with the Employee abstract role to be able to update their employee profile and picture. For CRM, you must also provision users with the Resource abstract role, this permits users to be assigned to work on leads, opportunities, and other CRM work.

• Duty roles

Job and abstract roles permit users to carry out actions by virtue of the duty roles they include. For example, the Sales Manager job role includes the Sales Lead Follow Up Duty and the Quota Management Duty. The Sales Management Duty makes it possible for the managers to create and update a sales lead, qualify a sales lead, and convert a sales lead into an opportunity. The Quota Management Duty enables the management of sales territory quotas and territory quota formulas.

• Functional security policies

Duty roles include functional security policies that provide access to user interface elements, Web services, tasks flows, and other functions. For example, a sales manager who has the Delete Opportunity functional policy can view and click the Delete button. Removing that policy removes the button from view.

A functional policy is made up of the duty role name and the Delete Opportunity functional privilege. The functional privilege specifies the application features that are being secured. In the security reference manuals, functional privileges are listed in the Privileges section.

• Data security policies
Duty roles also include data security policies that specify which roles can perform an action under what conditions. For example, the Opportunity Sales Manager Duty includes a data security policy that specifies that sales managers can view opportunities if they are in the management chain or are members of the sales team on the opportunity.

Each data security policy represents an underlying SQL query. Oracle Fusion CRM Applications evaluate the query at run time, and permit access to data that meets the condition.

A data security policy is composed of the name of the duty where it applies, a data privilege, and a condition. A data privilege is the combination of the action users can take, the conditions under which they can carry them out, and the object they can act on. Data privileges are listed in the Data Security Policies section of the security reference manuals.

**Note**

Unlike other Oracle Fusion Applications, Oracle Fusion CRM Applications do not use data roles (not shown in this diagram) to provide users with data access. They rely strictly on data security policies.

Data roles, which inherit enterprise roles, are used in many Oracle Fusion Applications to restrict user access to a dimension of data, such as a business unit or a data reference set.

How enterprise roles work in practice is best illustrated with an example from the Sales Manager job role outlined in the following diagram:

- A provisioning rule automatically provisions employee sales managers with the enterprise roles they need to do their jobs.
- The rule provisions the Sales Manager job role and the Employee and Resource abstract roles.
- The Sales Manager job role includes the Quota Viewing Duty and the Sales Manager Duty.
- Duty roles inherit other duty roles. For example, the Sales Manager Duty inherits many other duty roles including the Marketing Lead Analysis Duty and the Opportunity Sales Manager Duty.
- The duty roles are associated with functional security policies and data security policies. For example, the inherited Opportunity Sales Manager Duty comes with the following:
  - Functional security policies that specify which application pages and functions sales managers can access for deleting, assigning, closing, creating, and viewing an opportunity. For example, the view opportunity policy permits sales managers to view all UIs, Web services, and task flows that are related to opportunities.
  - Data security policies that specify what actions opportunity sales managers can take on what opportunities and under what conditions. For example, opportunity sales managers can view all data related to opportunities where they are opportunity sales team members with view, edit, or full access.
The following diagram provides more detail about the composition of a policy. Each policy, such as the View Opportunity policy, is composed of a duty role name and a privilege:

- The view opportunity functional security policy is composed of the duty name and the View Opportunity functional privilege.

- The view opportunity data security policy is composed of the duty name, the View Opportunity data privilege, and a condition. It specifies that sales managers can view all data related to opportunities where they are an opportunity sales team member with view, edit, or full access.
About the Security Reference Implementation

Details about the available enterprise roles, duty roles, and policies in the security reference implementation are described in reference manuals organized by business process.

Each of the enterprise roles provided by Oracle is composed of a hierarchy of other roles and duties. The following diagram displays a portion of the hierarchy for the Sales Manager job role from the Oracle Fusion Applications Sales Security Reference Manual as an illustration.
Sales Manager Job Role

| Business Intelligence Applications Worker |
| Business Intelligence Applications Analysis Duty |
| Quota Viewing Duty |
| Sales Manager Duty |
| Appointment Note Management Duty |
| Appointment View Duty |
| Asset and Recommendation View Duty |
| Incentive Compensation Participant Management Duty |
| Incentive Compensation Participation Duty |
| Incentive Compensation Transaction Analysis Duty |
| Business Intelligence Authoring Duty |
| Interaction Management Duty |
| Internal Contact Relationship Information Inquiry Duty |
| Lead Registration Processing Duty |
| Locations Management Duty |
| Locations Inquiry Duty |
| Marketing Lead Analysis Duty |
| Business Intelligence Authoring Duty |
| Mobile CRM Sales Representative Duty |
| Opportunity Sales Account Sales Manager Duty |
| Opportunity Sales Account Territory Resource Duty |
| Opportunity Sales Manager Duty |

The Sales Manager job role inherits the Business Intelligence Applications Worker abstract role. This abstract role comes with the Business Intelligence Applications Analysis Duty that permits the viewing of business analysis reports.

The Sales Manager job role also inherits the Sales Manager duty role. All job roles include a top-level duty role with the same name as the job role. This top-level duty role is the container for all of the duty roles assigned to a job role. As a general rule, the permissions are attached to the duties at the lowest level of the hierarchy.

The top-level duty roles make it easier for you to create job roles of your own. For example, if you want to create a new job role because you want to give additional functionality to a special class of sales managers, then you can assign the top-level Sales Manager duty to the new job role to give the users all the same permissions as a regular sales manager and then add whatever additional duty roles you want. (For an example of how to do this task, see the Enabling Salespeople to Obtain Microsoft Outlook Access to All Sales Accounts: Worked Example help topic.)

How Opportunity Information Is Secured: Explained

This topic explains how the security reference implementation provided by Oracle determines who can access what opportunity information in your CRM organization.
Whether or not you can access a particular opportunity depends on your membership in the resource and territory hierarchies. You have access the opportunity if:

- You create the opportunity.
- You are on the opportunity sales team.
- You are the owner of the territory.
- The opportunity owner is your direct or indirect report in the resource hierarchy.
- If you are assigned to a territory for the sales account associated with the opportunity or if a territory is assigned to the revenue lines on the opportunity.

Access differs between territory members and opportunity members:

- An opportunity owner gets full access to the opportunity, which includes the ability to edit as well as add and remove team members.
- Owners and members of territories assigned to the sales account of the opportunity get read-only access to the opportunity and are not added to the opportunity sales team.
- Owners and members of territories assigned to the opportunity revenue lines are added as a distinct list of territories to the opportunity sales team. Owners and members of these territories get full access to the opportunity. Depending on a profile option, either only the owner or all the members of the territory are added as resources to the opportunity sales team. Regardless of the access level for these members as a resource on the opportunity team, they will always have full access.

The following figure illustrates some of the different ways you gain access to an opportunity:

- Named agents in the diagram (A, B, and C) can access the opportunity
- Unnamed agents (highlighted in yellow) cannot access the opportunity
- Sales managers can access the opportunity because a salesperson in their management chain has access.

This figure shows who in a sales hierarchy can access an opportunity:
Agent A can access the opportunity because she created it. When you create an opportunity, you are the initial owner.

Agent B can access the opportunity because he is on the sales team.

Agent C can access the opportunity because he is the owner of the NW territory.

Sales managers who are higher up in the management chain can also see the opportunity because access is provided through the resource hierarchy. Agent C’s manager can access the opportunity information, but agent C’s colleagues cannot.

Note
Access using revenue lines and sales accounts are not shown in this figure.

Special Access

Some access is not affected by the management hierarchy and membership in sales teams or territories. This special access includes:

- Administrators: Administrators get access to opportunities and other objects. This access is based on their privileges, regardless of where the administrators are in the management hierarchy. Administrators do not have to be on the sales team or members of territories.

- Deal Protection: Salespersons assigned to an opportunity retain the sales credit on an opportunity even if they are moved to another opportunity.
Team Selling: You can configure the application to allow salespersons to see all opportunities related to their sales accounts.

**Oracle Fusion CRM Security: Architectural Overview**

Setting up Oracle Fusion CRM security involves the use of multiple application components outlined in this topic. For example, you create duty roles and enterprise roles in separate Oracle Fusion Middleware applications and the rules to provision them in Oracle Human Capital Management (HCM). While you can access all of these application components from the *Setup and Maintenance* work area, understanding what components are used for what purpose and terminology differences in these applications will help you with your setup.

**Application Components Used for Implementing Oracle Fusion CRM Security**

This diagram provides an overview of the key security setups and the application components that you use to configure them.

---

The following table provides an explanation of the application components:

<table>
<thead>
<tr>
<th>Application Component</th>
<th>What It Is Used For</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common CRM Setups</td>
<td>Tasks such as creating resource roles and resource organizations are part of common CRM configuration tasks.</td>
</tr>
<tr>
<td>Oracle Fusion Human Capital Management (HCM)</td>
<td>Oracle Fusion HCM tasks permit you to manage CRM application users and create the rules that automatically provision users with the enterprise roles they need to do their jobs.</td>
</tr>
<tr>
<td>Oracle Authorization Policy Manager (APM)</td>
<td>You use Oracle Authorization Policy Manager to create or customize duty roles and functional or data security policies in a separate browser window. The Authorization Policy Manager is a separate Oracle Fusion Middleware application with many features that are not used in Oracle Fusion CRM. You can access this application from the <strong>Setup and Maintenance</strong> work area by using the <strong>Manage Data Security Policies</strong> task.</td>
</tr>
<tr>
<td>Oracle Identity Management (OIM)</td>
<td>Use Oracle Identity Management to create new job and abstract roles for on-premise applications in a separate browser window. You can access this Oracle Fusion Middleware application from the <strong>Setup and Maintenance</strong> work area by using the <strong>Manage Job Roles</strong> task. OIM is a separate Oracle Fusion Middleware application with few features used by Oracle Fusion CRM. Because this application has not been extended to Oracle Sales and Marketing Cloud Service, you must ask your Service Administrator to create any new job roles for you. <strong>Note</strong> Creating a user in Oracle Identity Management is different from creating CRM application users in HCM.</td>
</tr>
<tr>
<td>Application Access Controls Governor (AACG)</td>
<td>The Governance, Risk, and Compliance Controls supports segregation of duties (SOD) using the Application Access Controls Governor. You can use the Application Access Controls Governor to manage application access controls designed to prevent conflicts of interest and potential fraud that could result from the duties you assign to job roles. For example, if you want to prevent the same users from being able to both create items and ship them to customers, then you can review the access controls provided by Oracle in the Segregation of Duties Policies Respected section of the security reference guides. When you assign incompatible duties to a job role that you are creating, the Application Access Controls Governor generates error messages and prevents the assignment.</td>
</tr>
</tbody>
</table>
Lightweight Directory Access Protocol (LDAP) Store

The security settings you create are stored in the Lightweight Directory Access Protocol (LDAP) store for quick access.

If you are implementing Oracle Fusion CRM on premise, then you must ensure that you synchronize the LDAP store with any of the changes you make by using the Run User and Roles Synchronization Process task, available from the Setup and Maintenance work area. If you are implementing Oracle Fusion CRM in the Oracle Sales and Marketing Cloud Service, then this step is done for you.

Terminology Differences Between Applications

Oracle Fusion CRM applications use different terminology from Oracle Fusion Middleware applications. The key differences are described in the following table:

<table>
<thead>
<tr>
<th>Oracle Fusion CRM Term</th>
<th>Definition</th>
<th>Equivalent Oracle Identity Management Term</th>
<th>Equivalent Oracle Authorization Policy Manager Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job role</td>
<td>Job roles permit users to perform activities required for their job.</td>
<td>Role. You can distinguish preconfigured job roles by their technical names. Job roles include the term JOB at the end.</td>
<td>External role. You can distinguish preconfigured job roles by their technical names. Job roles include the term JOB at the end.</td>
</tr>
<tr>
<td>Abstract role</td>
<td>Abstract roles permit users to perform functions that span the different jobs in the enterprise.</td>
<td>Role.</td>
<td>External role.</td>
</tr>
<tr>
<td>Duty role</td>
<td>Duty roles provide all the privileges for the actions required to carry out a job.</td>
<td>Not applicable.</td>
<td>Application role. Duty roles include the term DUTY at the end of the technical name.</td>
</tr>
</tbody>
</table>

Oracle Fusion CRM Security: Implementation Planning

Users gain access to application functionality when you provision them with enterprise roles. Oracle provides you with the enterprise roles required for all the standard jobs in a CRM organization. This topic outlines the steps you must complete to provision users with the enterprise roles provided by Oracle and what additional steps you must complete if you are planning to make changes.

Implementing Security If You Plan to Use Enterprise Roles Provided by Oracle

Follow these steps if you want to provision users with the preconfigured enterprise roles provided by Oracle:
1. Create resource roles if the existing resource roles do not match the job titles in your organization.

Resource roles indicate the role the resource plays in the CRM organization. When you create users for Oracle Fusion CRM applications, you must specify their resource role, which appears as the job title of the person in the resource directory and in social applications, such as the Activity Stream.

You also use resource roles to trigger the provisioning rules you will set up in the next step.

Oracle supplies the resource roles that correspond to the preconfigured job roles. For example, Oracle includes the Sales Vice President and the Marketing Vice President resource roles, which correspond to the Sales VP and the Marketing VP job roles.

If your organization uses different titles, then you must create additional resource roles.

You can review a list of the existing resource roles and create additional resource roles by navigating to the Setup and Maintenance work area and searching for the Manage Resource Role Lookups task.

See the Creating Resource Roles: Worked Example help topic for step-by-step instructions on creating resource roles.

2. Create rules that automatically provision users with job roles and abstract roles.

When you create users, the provisioning rules automatically provision users based on the user resource role and employment status.

You must create the provisioning rules from the Setup and Maintenance work area, by using the Manage HCM Role Provisioning Rules task.

See the following help topics for information on best practices to create the rules and step-by-step instructions:

- Rules to Automatically Provision Oracle Fusion CRM Users with Enterprise Roles: Explained
- Creating Rules to Automatically Provision Enterprise Roles to Oracle Fusion CRM Users: Worked Example

3. You are now ready to create users.

**Implementing Security If You Plan to Customize Enterprise Roles Provided by Oracle**

The following steps will help you plan your security implementation if you decide to make changes to the preconfigured job roles and abstract roles provided by Oracle:

1. Review the enterprise roles provided by Oracle.

Details on the available job roles, abstract roles, duty roles, privileges, and data security roles are described in reference manuals organized by
Define Common Applications Configuration for Marketing

2-121

different business areas, such as sales, marketing, and partner relationship management.

If the jobs in your organization do not match the job roles provided by Oracle, then you can modify access for users by assigning them with multiple existing job roles or creating new job roles. Creating new job roles makes it possible for you to select which duty roles to include and, in some cases, to configure new data security policies that govern on what data those duty roles grant access to.

2. Create any additional job roles.

You create additional job roles in Oracle Identity Manager, an Oracle Fusion Middleware application, which can be accessed from the Setup and Maintenance work area by using the Manage Job Roles task.

---

Note

If you are implementing security in the Oracle Sales and Marketing Cloud Service, then you must request the Oracle provision administrator to create job roles for you because access to this functionality has not been extended to the Oracle Sales and Marketing Cloud Service.

---

Before creating job roles, review the hierarchy of job roles and duty roles in the security reference manuals. The hierarchy of duty roles is not visible when you view them in Oracle Identity Manager.

Each job role includes a top-level duty role with the same name, which inherits all the other duty roles. If you want to give additional functionality to a group of users, for example, sales managers, then assign the Sales Manager Duty role to the job role you are creating to inherit all the current permissions and data access, and then add whatever additional duty roles you needed.

---

Tip

Oracle Identity Manager uses different terminology from Oracle Fusion CRM. Job roles and abstract roles are referred to as roles.

---

3. Create duty roles and add them to the job roles that you created.

If you do create a job role, then use the Oracle Authorization Policy Manager, which is a separate Oracle Fusion Middleware application, to do the following:

a. Create a top-level duty role for the job role.

b. Add the duty roles that you want to inherit to that top-level duty role.

c. Associate the job role that you created with your duty role on the External Role Mapping tab.

---

Tip
Oracle Authorization Policy Manager uses different terminology from Oracle Fusion CRM. Duty roles are referred to as application roles and job roles as external roles. You can identify duty roles by the word DUTY at the end of their names.

You can access Oracle Authorization Policy Manager from the **Setup and Maintenance** work area using the **Manage Duties** task.

The duty role that you create inherits all of the data security policies from the duty roles that you add. You can create additional data security policies if you want to change user access to specific data. For example, the predefined data security policies for the duty roles inherited by the Sales Representative Duty permit salespeople to view only sales accounts if they are on the sales account team. If you want to give salespeople access to all sales accounts, then you can create a data security policy.

4. Retrieve any security changes made in Oracle Identify Manager (stored in the Lightweight Directory Access Protocol (LDAP) store) so they are available for creating users.

Modifications to the reference become available only after the LDAP store is synchronized. The Oracle Sales and Marketing Cloud Service takes care of the synchronization for you. If you are implementing on premises, then you must run the synchronization process from the **Setup and Maintenance** work area using the **Run User and Roles Synchronization Process** task.

5. Follow the steps for implementing the preconfigured security described in the Implementing Security If You Plan to Use Enterprise Roles Provided by Oracle section.

### Oracle Fusion CRM Security: Special Considerations

This topic explains what happens if you omit the security setups and highlights the differences between implementing security on premises and in the Oracle Sales and Marketing Cloud Service.

**Minimum Required Security Setups**

If you plan to use only the preconfigured enterprise roles provided by Oracle, then you must:

1. Review the existing resource roles to see if they correspond to the titles of the different resources in the CRM organization, and create any additional resource roles you need.

2. Create the provisioning rules to automatically provision the required job roles and abstract roles.

3. If you are implementing on premises, then you must Synchronize the Lightweight Directory Access Protocol (LDAP) store by using the **Run User and Roles Synchronization Process** task from the **Setup and Maintenance** work area.
If you are implementing in the Oracle Sales and Marketing Service, then the process is run for you.

**Differences Between Implementing Security On Premises and In the Cloud**

The following table highlights the differences between implementing security on premises and in the Oracle Sales and Marketing Cloud Service:

<table>
<thead>
<tr>
<th>Task</th>
<th>On Premises</th>
<th>Oracle Sales and Marketing Cloud Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating additional job roles</td>
<td>You can create additional job roles from the <strong>Setup and Maintenance</strong> work area using the <strong>Manage Job Roles</strong> task.</td>
<td>You must request the service administrator to create additional job roles.</td>
</tr>
<tr>
<td>Synchronizing the LDAP store</td>
<td>You must synchronize any changes to security by using the <strong>Run User and Roles Synchronization Process</strong> task from the <strong>Setup and Maintenance</strong> work area.</td>
<td>This process is run for you.</td>
</tr>
</tbody>
</table>

**Siebel CRM and Oracle Fusion CRM Security Differences: Explained**

Siebel CRM and Oracle Fusion CRM implement different methods of securing access to application functionality and data. This topic outlines the mechanisms provided by Siebel CRM to control the privileges or resources that users are entitled to after they have accessed a Siebel application and been authenticated. For additional information on Siebel CRM security, see Siebel Security Guide on Oracle Technology Network.

Siebel CRM uses two primary access-control mechanisms:

- View-level access control to manage the application functionality that a user can access
- Record-level access control to manage the data items that are visible to each user on a view

**View-Level Access control**

In Oracle Fusion CRM, access to functionality is provided by assigning enterprise roles to users. In Siebel CRM, access to functionality is provided by assigning responsibilities to users.

Within each Siebel application, screens provide a broad area of functionality. A screen is composed of views, and the collection of views to which users have access determines the application functionality available to them. Access to views is determined by responsibilities.

Organizations are generally arranged around job functions, with employees being assigned one or more functions. In Siebel CRM, these job functions are called responsibilities. Each responsibility is associated with more or more views, which represent data and functionality needed for a job function. Each user must be assigned at least one responsibility to access the Siebel application.

Siebel Business Applications ship with many predefined responsibilities. However, you can also define any additional responsibilities that you require that correspond to the major job functions in your organization.
Record-Level Access Control

Record-level access control is used to assign permissions to individual data items within an application so that only authenticated users who need to view particular data records have access to that information. In Oracle Fusion CRM, access to data is primarily determined by the data security policies that apply to specific enterprise roles. In Siebel CRM, the access control mechanism that applies to a view determines the data records that a user sees in a view.

Siebel CRM uses the following types of record-level access control mechanisms:

- Personal access control. Access to data associated with a user’s Person record in the database is restricted to that user.
- Position access control. Access to data that is specific to a job title is restricted to users assigned that job title, to their team members, or to their subordinates.
- Organization access control. Employees assigned to a position within an organization are granted access to the data assigned to that organization.
- Access-group access control. Access groups are used to control access to master data by diverse groups of individuals.
- All-access control. All-access control provides access to all records in a view that has a valid owner. If a view applies all-access control, then all users with access to the view see the same data in the view.

Business Components and Record-Level Access Control

Within Siebel CRM, views are based on business components and must use one of the view modes specified for the business component. A business component’s view mode determines which access control mechanisms can be applied to the business component in any view. Applet and view properties also determine the data available in a view. Applet visibility properties define the business component on which a view is based, and a view’s access control properties determine what access control mechanism is applied to the business component on which the view is based. For example, a business component might have personal or position access control available. The view access control property specifies which of these to use.

Creating Resource Roles: Worked Example

Follow the steps in this topic to create resource roles. Resource roles, for example, Sales Manager, Salesperson, or Vice President of Marketing, describe the role that a resource plays in the CRM organization and appear as job titles in the resource directory and in social applications, such as Activity Stream. Resource roles are also used to assign users with the enterprise roles they need to carry out the duties of their job.

After you create a resource role, you must create the appropriate provisioning rules to provision the user with the required enterprise roles. The resource role by itself is only a title.

Note

Common CRM resource roles are already set up for you. These are labeled as System roles in the application. To obtain a list, click Search in the Manage Resources page without entering any search criteria.
Steps to Create a Resource Role

1. Navigate to the Setup and Maintenance work area by selecting the link in the Navigator menu.
2. On the All Tasks tab, search for the Manage Resource Role task.
3. Click the Go to Task button.
   The Manage Resource Roles page appears.
4. Click the Create button.
   The Create Resource Role page appears.
5. In the Role Name field, enter the name of the resource role as it will appear in the application, for example, CEO.
6. In the Role Code field, enter a unique internal name. No spaces are permitted.
7. Select the Manager option if the resource role belongs to a manager, or the Member option if the resource role belongs to a single contributor.
8. In the Role Type list, select either Sales or Marketing to classify the role that you are creating. Your selection has no impact on the security functionality.
9. Click the Save and Close button.

Rules to Automatically Provision Oracle Fusion CRM Users with Enterprise Roles: Explained

By creating rules using the Manage HCM Role Provisioning Rules task from the Setup and Maintenance work area, you can automatically provision users with all the enterprise roles they need for their job. These roles ensure the users have access to all the application functions and data they need to carry out their job duties. The rules automatically provision the users when they are created or move positions within your organization. If resources leave your company, the roles can be automatically removed so the resources can no longer access the application.

You must ensure that the rules you create assign Oracle Fusion CRM application users with:

1. One or more job roles required to perform their job in the organization.
2. If the users work in the CRM organization, then you must assign them the Resource abstract role. This role permits users to be assigned to sales teams, territories, and other CRM work.
3. If the users are employees, then you must also assign them with the Employee abstract role. If they are temporary workers, then you must instead assign them with the Contingent Worker abstract role.

In CRM, both abstract roles assure that users can update their personal profiles and other common tasks.

As a best practice:
• Create rules that provision the Resource abstract role and the job roles users need to perform their job based on the resource role assigned to them. Most resource roles will require just one job role, but some may have more than one. For example, the Sales and Marketing Vice President may require both the Sales VP and the Marketing VP job roles.

• Create one rule that provisions all users who are employees (Person Type of Employee) with the Employee abstract role.

• If you hire contractors, then create a separate rule to provision the Contingent Worker abstract role to all users with the Person Type of Contingent Worker.

• For all rules you create, ensure that the privileges the users are granted are automatically removed when they are no longer with the company. You can do this by stipulating the condition that the rule apply only when the user has the Assignment Status of Active.

• If you plan to create users to perform implementation tasks who are not resources in the CRM organization, then you must create a provisioning rule that is based not on the resource role but some other user attribute. For example, you can create a rule to provision the required Application Implementation Consultant and IT Security Manager job roles based on the user’s job assignment.

The following figure provides an example of the rules you would create to provision employee users who are assigned the Sales Manager resource role:

• Rule one assigns the Employee abstract role to users who are employees with an Assignment Status of Active.

• Rule two assigns the Sales Manager job role and the Resource abstract role to users with the Sales Manager resource role and an Assignment Status of Active.
Creating Rules to Automatically Provision Enterprise Roles to Oracle Fusion CRM Users: Worked Example

Follow the steps in this example to create rules that automatically provision Oracle Fusion CRM application users with the necessary enterprise roles. The provisioning is based on the resource role that you assign to a user.

In this example, you create a rule to provision users with the Sales Vice President resource role with the enterprise roles they need to perform their jobs.

Steps to Create an Autoprovisioning Rule

1. From the Navigator menu, click the Setup and Maintenance link located under the Tools heading.

2. On the Overview page All Tasks tab, search for the Manage HCM Role Provisioning Rules task.

3. Click the Go to Task button for the Manage HCM Role Provisioning Rules task.

   The Manage HCM Role Provisioning page appears.

4. Click the Create button.

   The Create Role Mapping page appears.

5. In the Mapping Name field enter a name, for example, Sales Vice President.

6. In the Conditions region, enter the resource role as a condition. In this example, you enter Sales Vice President in the Resource Role field.

7. Enter Active for Assignment Status.

   This additional condition ensures that the provisioned enterprise roles are automatically removed if the user is terminated.

8. In the Associated Roles region, click Add to add the enterprise roles. For this example, you add the following:

   • Sales Vice President
   • Resource

   Note

   Each CRM resource who is an employee must be provisioned with both the Resource and Employee abstract roles. You must create a separate rule that assigns the required Employee abstract role to all users who are employees. You must always provision the Resource role along with the appropriate job roles. This provisioning ensures that the user can be assigned work in your CRM application.

9. Make sure the Autoprovion option is selected for all the job roles.

10. Click Save and Close.
Define Users for CRM

Creating Oracle Fusion CRM Application Users: Functional Overview

This topic outlines concepts that will help you understand and plan the creation of Oracle Fusion CRM applications users.

Types of Users That You Can Create

The types of users that are available to you differs for Oracle Sales and Marketing Cloud Service implementations and on-premises implementations.

- On-premises implementations can use Oracle Identity Manager, an Oracle Fusion Middleware application, to create and provision enterprise roles to implementation users outside Oracle Fusion CRM applications.

- Oracle Sales and Marketing Cloud Service implementations do not have access to Oracle Identity Manager, so you must create and provision implementation users within Oracle Fusion CRM. The term setup users distinguishes the implementation users created from within the Oracle Fusion CRM applications from those created in Oracle Identity Manager.

The following table lists the different user types. Because permissions granted to users depend on the enterprise roles you assign them, you are not restricted to the user types listed in the table.

<table>
<thead>
<tr>
<th>Type of User</th>
<th>Description</th>
<th>Available in On-Premises Implementations of Oracle Fusion CRM</th>
<th>Available in the Oracle Sales and Marketing Cloud Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Superuser (FAADMIN)</td>
<td>The initial superuser who sets up the Oracle Fusion Applications environment.</td>
<td>Yes, provided by Oracle after installation.</td>
<td>No.</td>
</tr>
<tr>
<td>Implementation Users</td>
<td>Yes.</td>
<td>No. Oracle Sales and Marketing Cloud Service implementations do not have access to Oracle Identity Manager.</td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td>------</td>
<td>------------------------------------------------------------------</td>
<td></td>
</tr>
</tbody>
</table>
| The FAADMIN superuser can create other implementation users outside Oracle Fusion CRM applications using Oracle Identity Manager. These implementation users complete the enterprise setup required for Oracle Fusion Applications and are enabled to manage user security and carry out DBA tasks, such as environment maintenance as well as creating and managing user accounts. Implementation users are not created as employees or resources in Oracle Fusion CRM, so you cannot assign them CRM application job roles. They cannot view CRM transaction data or reports. Implementation users are provisioned with the following enterprise roles:  
  - Application Implementation Consultant job role  
  - IT Security Manager job role |
| Setup Users                                                                 | Users with the same privileges as implementation users but who are created within Oracle Fusion Applications, using the same Create User page that is used to create other application users. Setup users can perform all of the same implementation setups for your CRM implementation, including managing security, setting up other users, and editing enterprise information. Setup users are not created as resources in Oracle Fusion CRM and are not provisioned with the Resource abstract role, so you cannot assign them CRM application job roles and they cannot view CRM transaction data or reports. Setup users are provisioned with the following enterprise roles:  
  - Application Implementation Consultant job role  
  - IT Security Manager job role  
  - Employee abstract role | Setup users are not provided for on-premises implementations. If you want to create a setup user, then you must create a provisioning rule to provision enterprise roles based on a property other than the resource role. See the Creating Setup Users for Oracle Fusion CRM: Worked Example help topic for details. | Oracle provides you with one initial user with the same access as a setup user based on the information you provided when you signed up with the service. If you want to create additional setup users, then you must create a provisioning rule to provision enterprise roles based on a property other than the resource role. See the Creating Setup Users for Oracle Fusion CRM: Worked Example help topic for details. |
| Sales Administrators | Sales administrators are CRM application users who are provisioned with the Sales Administrator job role. They can create other CRM application users, manage data import from legacy systems, and customize the application. Unlike setup users, sales administrator users can view CRM transactional data and reports. They cannot configure CRM application security or perform tasks related to the enterprise setup. Sales administrator users are provisioned with the following enterprise roles: • Sales Administrator job role • Resource abstract role • Employee abstract role | Yes. | Yes. |
| CRM Application Users | Implementation users, setup users, and sales administrators can create CRM application users such as marketing managers and salespersons. Application users are provisioned using their role in the organization with the security settings they need to perform their jobs. They can perform only functional setup within the application, depending on their role. Application users are provisioned with the following enterprise roles:  
- The job roles required to perform their job  
- The Resource abstract role  
- The Employee or the Contingent Worker abstract role, depending on the status of the user as an employee or contractor | Yes. On premises implementations can create application users in any of three ways:  
- Manually one-by-one from the Manage Users task from the Setup and Maintenance work area  
- By importing them from a file using file-based data import  
- By loading users directly into database interface tables using a loader of your choice | Yes. Because the loader option is not available, Oracle Sales and Marketing Service Cloud implementations can create application users in one of two ways:  
- Manually one-by-one from the Manage Users task from the Setup and Maintenance work area  
- By importing them from a file using file-based data import |

**Ways of Creating Oracle Fusion CRM Application Users**

You can create Oracle Fusion CRM application users in multiple ways. You can:

- Create users individually in the user interface
  
  Use this method for creating implementation and sales administrator users and for creating application users unless the number of the number users you are creating is very large.

- Import users from a file
  
  Import users from a file using the file-based import feature only if you have a very large number. You cannot import implementation users because the import process requires you to import CRM resources.

- Load user records into the database tables
  
  If you are implementing CRM on premises and you are creating a large number of users, then you can load users into the application using a loader of your choice.
### Tasks That Are Accomplished When You Create Users

When you create CRM implementation and application users, you are accomplishing multiple tasks at the same time, depending on the type of user. The following table lists the tasks:

<table>
<thead>
<tr>
<th>Task Accomplished</th>
<th>CRM Application Users</th>
<th>Setup User</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sends automatic e-mail notifications with user names and automatically generated temporary passwords</td>
<td>Yes</td>
<td>Yes</td>
<td>The application sends the notifications to the user or to an administrator only once, either on creation or at a later time, depending on the setup.</td>
</tr>
<tr>
<td>Provisions the enterprise roles that provide the security settings that users need to do their jobs</td>
<td>Yes</td>
<td>Yes</td>
<td>Enterprise roles are provisioned based on the autoprovisioning rules you create as part of the security setup.</td>
</tr>
<tr>
<td>Creates resources that can be assigned CRM work</td>
<td>Yes</td>
<td>No</td>
<td>Only users created as resources can be assigned to sales teams and view reports.</td>
</tr>
<tr>
<td>Creates the resource reporting hierarchy used by Oracle Fusion CRM for reporting, forecasting, and work assignment</td>
<td>Yes</td>
<td>No</td>
<td>You create the hierarchy by specifying a manager for each resource.</td>
</tr>
<tr>
<td>Creates resource records that users can update with personal information to complete a directory of your CRM organization</td>
<td>Yes</td>
<td>No</td>
<td>Only resources have their information appear in the CRM directory.</td>
</tr>
<tr>
<td>Creates the hierarchy of resource organizations</td>
<td>Yes</td>
<td>Not applicable</td>
<td>Each resource is assigned to a resource organization. The application uses the resource reporting hierarchy to build a hierarchy of these organizations.</td>
</tr>
<tr>
<td>Creates rudimentary employee records for use by Oracle Fusion HCM.</td>
<td>Yes</td>
<td>Yes</td>
<td>All users you create in the user interface or by importing generate employee records.</td>
</tr>
</tbody>
</table>

### Setting Up E-Mail Notifications for New Users

For each CRM user that you create, you must enter a unique e-mail address. By default, the application automatically sends an e-mail notification with the user name and temporary password to this address immediately after the user is created. Users then sign in and change their passwords.
If you do not want users to receive the notification e-mail right away because you are in the trial phase of your implementation project, then you can disable the automatic notification using the following steps:

1. Navigate to the Setup and Maintenance work area.
2. Search for the task Manage Enterprise HCM Information on the All Tasks tab of the Overview page.
3. Click the Go to Task button.
4. In the Enterprise page, click the Edit button, and select Update.
5. In the User and Role Provisioning Information region, set the Send User Name and Password option to No.
6. Click Done.

When users are ready to receive their temporary passwords, you can send all of the notifications at the same time, using the following steps:

1. In the Navigator menu, select Scheduled Processes under the Tools heading.
2. In the Scheduled Processes Overview page, click Schedule New Process.
3. In the Schedule New Process dialog box, make sure the Job option is selected for Type.
4. Enter Send User Name and Password E-Mail Notifications in the Name field.
5. Click OK.
6. In the Process Details window, click Submit.
7. Click Close.

**Note**
The Send User Name and Password E-Mail Notifications process sends the notification e-mail only to those users who have never been sent the notification. The process does not reset passwords or resend the notification.

 Alternately, you can send a notification to an individual user:

1. While editing the user in the Create User or Edit User page, select the Send User Name and Password check box in the User Notification Preferences region.

**Creating the Resource Reporting Hierarchy**
The resource reporting hierarchy provides the basis for CRM data security. The resource reporting hierarchy need not mirror the formal reporting hierarchy, which is captured separately in the Oracle Fusion HCM application if it has been implemented.

**Note**
In Oracle Fusion CRM, you can have only one hierarchy reporting to one person.
You build a resource reporting hierarchy when you create CRM application users by specifying the manager for each user. If you are creating users one-by-one in the user interface, then you must start by creating the user at the top of the hierarchy and work your way down. If you are importing users using file-based import, then the order does not matter provided that all of your users are in the same file.

**Creating Resource Organizations and the Resource Organization Hierarchy**

In Oracle Fusion CRM, you must assign each manager that you create as a user with his or her own resource organization. All direct reports who are individual contributors inherit their manager's organization.

In Oracle Fusion CRM, resource organizations serve a limited purpose. Their names appear in the application's Resource Directory, which users can access to obtain information about their coworkers, and in social media interactions. Resource organizations are not used for work assignment.

The following screen capture shows the Resource Directory, which is available on the application Navigator. The resource organization names appear under each person's title.

The application automatically builds a resource organization hierarchy, using the resource reporting structure.

Suppose, for example, that your CRM enterprise includes sales and marketing departments that report to the Executive VP of Sales and Marketing and its members as follows:

- Mathew Fullerton, Executive VP Sales and Marketing
- Bob Doyle, Sales VP
- Mateo Lopez, Sales Manager
- Gabrielle Lee, Sales Manager
- Jillian Henderson, Sales Representative (reporting to Mateo Lopez)
- Joseph Kerr, Sales Analyst (reporting to Gabrielle Lee)

A diagram of the reporting hierarchy looks like the following:
Now, also suppose that you create the following resource organizations and assign them to the managers.

<table>
<thead>
<tr>
<th>Manager</th>
<th>Assigned Resource Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mathew Fullerton</td>
<td>Sales and Marketing</td>
</tr>
<tr>
<td>Bob Doyle</td>
<td>Sales</td>
</tr>
<tr>
<td>Lillian Jones</td>
<td>Marketing</td>
</tr>
<tr>
<td>Mateo Lopez</td>
<td>Sales West</td>
</tr>
<tr>
<td>Gabrielle Lee</td>
<td>Sales East</td>
</tr>
</tbody>
</table>

The application automatically builds the resource organization hierarchy, shown in the following figure, using the hierarchy of managers.

The resource organizations remain even if managers leave. You can reassign the resource organizations to their replacements.

The resource organization names do not have to reflect the names of departments. Departments are tracked along with employee records in the Oracle Fusion HCM application if it has been implemented. The resource organizations are not used in application security or to assign work to users. For example, you cannot include a resource organization on an opportunity sales team or as a territory owner.
Creating Oracle Fusion Human Resources Employee Records

When you create application users, you must include information that is used to create basic employee records for the Oracle Fusion HCM application. This requirement is part of the CRM application architecture. These records are used only if you are implementing this application now or plan to do so.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Definition</th>
<th>On Premises</th>
<th>Oracle Fusion Sales and Marketing Cloud Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Type</td>
<td>Enter either Employee or Contingent Worker, depending on whether the user you are creating is an employee or a contractor. The selection you make is used for provisioning either the Employee or the Contingent Worker abstract role.</td>
<td>Enter either Employee or Contingent Worker.</td>
<td>Enter either Employee or Contingent Worker.</td>
</tr>
<tr>
<td>Legal Employer</td>
<td>Enter the name of the legal entity that is the user's employer.</td>
<td>Enter the name of the legal entity that you defined as part of the enterprise setup.</td>
<td>Enter the legal entity that was set up for you based on the information you provided when you signed up with the service.</td>
</tr>
<tr>
<td>Business Unit</td>
<td>The business unit where your CRM applications are being used. All CRM applications must be implemented in the same business unit.</td>
<td>Enter the business unit that you defined as part of enterprise setup.</td>
<td>Enter the business unit that was set up for you based on the information you provided when you signed up with the service.</td>
</tr>
</tbody>
</table>

Creating Oracle Fusion CRM Application Users: Architectural Overview

This topic describes the different applications and application modules that you use when you create Oracle Fusion Customer Relationship Management (CRM) users. These include Oracle Fusion Human Capital Management (HCM), File-Based Data Import, Universal Messaging Service, and Oracle Identity Management, which is an Oracle Fusion Middleware application.

The following figure outlines the different application components that you use when you are creating and managing users:

- Creating users

  You can create CRM application users in one of three ways. Each method involves different application components.

  - To create users individually in the user interface, you select the Manage Users option in the application Navigator.
The Create User page is part of the Oracle Fusion HCM application, but you are not required to purchase or to implement Oracle Fusion HCM. The core Oracle Fusion HCM features that are required to create and manage CRM users are provided with Oracle Fusion CRM.

- To import users from a file, use the tasks in the setup task group Define File-Based Data Import. You can search for this task group from the Setup and Maintenance work area. You can use the File-Based Data Import task group, which is described in a separate topic, to import many application objects, including customer records, opportunities, and leads.

- If you are implementing CRM on premises, then you can also create users by importing them using a loader of your choice. To use a loader, select Data Import under the Customer Data Management heading in the Navigator menu, or use the Define Trading Community Import task list in the Setup and Maintenance work area.

- Managing notifications

  When you create users using any of these methods, you can automatically send an e-mail notification with the user name and temporary password. The e-mails are sent by the User Messaging Service, a part of the Oracle Service-Oriented Architectures Suite, which is a separate Oracle Fusion Middleware application. If you are implementing your CRM application on premises, then a superuser can also manage the notifications and the notification templates from the User Messaging Service application.

- Managing users

  When you create users, basic user information is copied to Oracle Identity Management, which is a separate Oracle Fusion Middleware application. If you are implementing CRM on premises, then you can use Oracle Identity Management for user management. For example, you can use Oracle Identity Management to reset user passwords and manage the enterprise roles that provision users with the security settings they require to carry out their duties. On premises implementations can use Oracle Identity Management to create implementation users, but not CRM application users.
The following table provides a breakdown by task or component:

<table>
<thead>
<tr>
<th>Component or Application</th>
<th>Description</th>
<th>Navigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define File-Based Data Import task group</td>
<td>Use the Define File-Based Data Import task group, which is a common component of Oracle Fusion Applications, to import users from a file.</td>
<td>Navigate to the Setup and Maintenance work area and search for the Define File-Based Data Import task group.</td>
</tr>
<tr>
<td>Oracle Fusion HCM</td>
<td>Use the Manage Users task to create and manage user records.</td>
<td>Select the Manage Users link in the Navigator.</td>
</tr>
<tr>
<td>User Messaging Service</td>
<td>If you are implementing CRM on premises, then you can use this application to manage the e-mail server that sends notifications to users. The notifications contain the user names and temporary passwords.</td>
<td>Sign in to the User Messaging Service as described in that application's documentation.</td>
</tr>
<tr>
<td>Oracle Identity Management</td>
<td>If you are implementing CRM on premises, then you can use Oracle Identity Management to create and manage job roles, as well as manage users and passwords. Although you can create users, you cannot create users as resources. This restriction means that you cannot create Oracle Fusion CRM application users in Oracle Identity Management alone.</td>
<td>You can use the Manage Job Roles task from the Setup and Maintenance work area to start the application in a separate browser window.</td>
</tr>
</tbody>
</table>
Creating Oracle Fusion CRM Application Users: Implementation Planning

This topic provides an overview of how to create Oracle Fusion Customer Relationship Management (CRM) application users and their resource organizations.

This topic covers:

- Creating the resource organization at the top of the CRM organizational hierarchy
- Creating the resource organizations that you must assign to each manager
- Creating the users

Prerequisites

Before you create CRM users you must:

- Make sure that all enterprise information has been set up.
- Complete the security setup. Make sure you have all the resource roles, enterprise roles, and autoprovisioning rules you need for the users you are creating.

Creating the Resource Organization at the Top of the CRM Hierarchy

Before you can create resource organizations for your users, you must create the top-level organization in your hierarchy.

To create the top-level organization, do the following:

1. Create the top-level organization as you would create any other resource organization by using the Manage Sales and Marketing Organizations task from the Setup and Maintenance work area.

2. Specify the resource organization that you created as the top of your sales and marketing hierarchy by using the Manage Resource Organization Hierarchies task from the Setup and Maintenance work area.

For more information, see the Creating the Top Level of the CRM Resource Organization Hierarchy: Worked Example help topic.

Creating Resource Organizations for Manager Users

If you are creating users individually in the user interface using the Create User page, then you can create the resource organizations while you are creating the users, or you can create them before using the Manage Sales and Marketing Organizations task from the Setup and Maintenance work area.

If you are importing users, then you must have the resource organizations created before you import any data.

Note

When creating resource organizations:

- Do not use the name of the manager in the name of your resource organization because you can reassign the organization to a new manager if the current one changes or leaves.
• Create a separate resource organization for each manager user. This requirement applies even if the managers work in the same organization.

For more information, see the Creating Resource Organizations for Oracle Fusion CRM: Worked Example help topic.

Creating Users

This section provides an overview of the different methods of creating Oracle Fusion CRM users:

• To create users individually in the user interface:
  
  Navigate to the Manager Users work area, which is available as a selection on the application’s Navigator.

The Create User page, displays up to six regions, depending on your release version, your enterprise defaults, and the enterprise roles assigned to you. The following screen capture shows the page.

The following table describes the regions.

<table>
<thead>
<tr>
<th>Create User Page Region</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Details</td>
<td>You are required to enter the user’s last name, hire date, and a unique e-mail address for the initial e-mail notification.</td>
</tr>
<tr>
<td>User Details</td>
<td>Enter the user name that the user will use to sign in.</td>
</tr>
<tr>
<td>User Notification Preferences</td>
<td>Select the option in this region if you want to send the user name and password to the user you are creating. This region appears only if you decided to turn off automatic notifications by setting the <strong>Send User Name and Password</strong> option to No while editing the enterprise information using the Manage Enterprise HCM Information task.</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Employment Information</td>
<td>The information you enter in this region is used to create the basic employee record for use with the Oracle Fusion Human Capital Management (HCM) application. Only the required fields are relevant to CRM.</td>
</tr>
<tr>
<td>Resource Information</td>
<td>Make entries in this region if you are creating a CRM application user who is a CRM resource. Leave this region blank for setup users who are not members of the CRM organization. You must enter a resource role and a manager for every CRM application user, because access to application data depends on the resource hierarchy. For every user with a manager resource role, you must enter a resource organization. The resource role you select determines the job roles that the user will be assigned when you click the Autoprovission Roles button.</td>
</tr>
<tr>
<td>Roles</td>
<td>Clicking the <strong>Autoprovission Roles</strong> button provisions the user with the enterprise roles that provide access to the application functions and data. If you are importing resources from a file, then the autoprovissioning is done automatically.</td>
</tr>
</tbody>
</table>

- To import users from a file:

Use the Manage File Import Activities task from the Setup and Maintenance work area, and create an import activity. To import users, you must select the Employee Resource object when creating the import activity. Before you can import users, you must:

- Understand what attributes are required, and what values are permitted.
- Create a mapping between the user data that you are importing and the attributes in the application.
Refer to Define File-Based Data Import topics for more information on how to import users and other data.

- If you are implementing CRM on premises, then you can load user data into the application by using a loader of your choice.

Use the Import Worker Users task from the Setup and Maintenance work area.

### Creating Oracle Fusion CRM Application Users: Special Considerations

This topic summarizes the main differences between creating users when you are implementing Oracle Fusion CRM on premises and in the Oracle Sales and Marketing Cloud Service.

The following table summarizes the main differences.

<table>
<thead>
<tr>
<th>Feature</th>
<th>On Premises</th>
<th>Oracle Sales and Marketing Cloud Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Oracle Identity Manager to create and manage users.</td>
<td>Yes.</td>
<td>No.</td>
</tr>
<tr>
<td>Loading users directly into interface tables using a loader of your choice</td>
<td>Yes.</td>
<td>No.</td>
</tr>
</tbody>
</table>

### Creating Resource Organizations for Oracle Fusion CRM: Worked Example

In Oracle Fusion CRM, you must assign a resource organization to each manager user that you create. All direct reports of that manager inherit the organization. You can create resource organizations before you create users according to the steps in this example.

**Steps to Create a Resource Organization**

1. Navigate to the Setup and Maintenance work area, and search for the Manage Sales and Marketing Organizations task.
2. Click the Go to Task button.
   
   The Manage Sales and Marketing Organization page appears.
3. Click the Create button.
   
   The Create Organization: Select Creation Method page appears.
4. Select Option 2: Create New Organization.
5. Click Next.
   
   The Create Organization: Enter Basic Information page appears.
6. Enter the name of the organization, for example, Sales.
7. In the **Organization Usages** region, click **Add Row** (green plus sign) to specify whether the organization is going to be used as a sales organization, a marketing organization, or both.

Specifying a usage determines whether the organization is visible when creating a sales manager or a marketing manager.

8. Click **Finish**.

**Creating the Top Level of the CRM Resource Organization Hierarchy: Worked Example**

Use this example as a guide for creating the resource organization at the top of the resource organization hierarchy. You must complete this setup before you create users.

When you create users who are managers in the Oracle Fusion CRM organization, you must assign a resource organization to each of them. The application automatically builds a resource organization hierarchy for you from the management hierarchy that you create.

Before you can create the resource organizations for the managers, you must create the top-level resource organization in your hierarchy following the steps in this example.

Creating the top-level resource organization in the resource organization hierarchy involves the following:

1. Creating the resource organization
2. Specifying the organization as the top of your sales and marketing hierarchy

**Creating the Resource Organization at the Top of the Hierarchy**

1. Search for the **Manage Sales and Marketing Organizations** task in the **Setup and Maintenance** work area.

2. Click the **Go to Task** button.

   The **Create Organization: Select Creation Method** page appears.

3. Click **Create** and select **Option 2: Create New Organization**.

4. Click **Next**.

5. Enter Global Organization (or another name of your choice) in the **Name** field.

6. In the **Organization Usages** region, click **Add Row** to add **Sales Organization** and **Marketing Organization**.

   This step identifies how the organization is being used.

7. Click **Finish**.

**Specify the Organization as the Top of Your Sales and Marketing Hierarchy**

1. Search for the **Manage Resource Organization Hierarchies** task in the **Setup and Maintenance** work area.
2. Click the Go to Task button. The Manage Resource Organization Hierarchies page appears.

3. Click Search.

4. In the Search Results, select the Sales and Marketing Organization Hierarchy link.
   You will associate the resource organization you created with the predefined hierarchy type.

5. From the Action menu at the top right-hand corner of the page, select Edit This Hierarchy Version.
   The Edit Organization Hierarchy Version page appears.

6. Click the Add button in the Global Sales and Marketing Organization Hierarchy region.
   The Add Tree Node window appears.

7. Click Search.
   The Search Node window appears.

8. Click Search.

9. Select Global Organization.

10. Click OK.
    The application returns you to the Edit Organization Hierarchy Version page.

11. Click Done.

Creating Setup Users for Oracle Fusion CRM: Worked Example

Follow the steps in this example to create a setup user, an application user with the privileges to set up application security, make changes to the enterprise setup, create other users, and complete most setup tasks. Because setup users are not created as CRM resources, they cannot view CRM data or reports and cannot be assigned to sales teams and territories.

Creating a Setup User Overview

Setup users do not work in the CRM organization and so should not appear in the CRM application directory or be available as resources for assignment to sales teams. For this reason, you do not specify a resource role or any other resource information for this type of user.

To provision these users with the required Application Implementation Consultant and IT Security Manager job roles, you must create a provisioning rule that is triggered not by the resource role assigned to the user, but by some other attribute. In this example, you will trigger the rule on the user's job.

To create the setup user, you will:

1. Sign in as a user with implementation privileges. This can be another setup user.
2. Create the job that will trigger the rule, Customer Administrator in this example.

3. Synchronize the Lightweight Directory Access Protocol (LDAP) directory with any changes to the security setups by running a scheduled process. The LDAP directory enables quick access by CRM Fusion Applications to security settings, so it is a good idea to update the directory to make sure it reflects all the latest security changes.

4. Create the provisioning rule to provision the required enterprise roles to users with the Customer Administrator job.

5. Create a rule to provision the Employee abstract role to all employee users. This is a one-time setup, so you can ignore this step if you created the rule previously.

6. Create the user.

Creating a Job
To create the job:
1. Navigate to the Setup and Maintenance work area.
2. On the All Tasks tab search for the Manage Job task.
3. Click the Go to Task button for the Manage Job task.
   The Manage Job page appears.
4. Click the Create button.
5. Enter a name for job that will be displayed for selection in the Create User page. For example, Customer Administrator.
6. In the Code field, enter an internal code for the job, for example, CUSTADMIN. The code must be uppercase with no spaces.
7. For Job Set, select Common.
8. Click Next.
   The Create Job: Details page appears.
9. Click Submit.

Updating the LDAP Directory
To update the LDAP directory:
1. Select the Scheduled Processes link in the Navigator.
2. Click the Schedule New Process button.
   The Schedule New Process window appears.
3. Select Retrieve Latest LDAP Changes from the Name list.
4. Click OK.
5. Click Submit.

Creating the Role Provisioning Rule for the Setup User
To set up the provisioning rule that automatically assigns the appropriate enterprise roles:
1. Navigate to the **Setup and Maintenance** work area.

2. On the **All Tasks** tab search for the **Manage HCM Role Provisioning Rules** task.

3. Click the **Go to Task** button for the **Manage HCM Role Provisioning Rules** task.

   The **Manage HCM Role Provisioning** page appears.

4. Click the **Create** button.

5. In the **Mapping Name** field enter a name, for example, Customer Administrator.

6. In the **Conditions** region, enter the job you created earlier as a condition. In this example, you would enter Customer Administrator in the **Job** field.

   Enter **Active** for **Assignment Status**. This additional condition ensures that the provisioned enterprise roles are automatically removed if the user is terminated.

7. In the **Associated Roles** region, use the **Add** button to add the following job roles:
   - Application Implementation Consultant
   - IT Security Manager

8. Make sure the **Autoprovision** option is selected for both job roles.

9. Click **Save and Close**.

**Creating the Role Provisioning Rule for All Employees**

You must create one rule to provision all users who are employees with the Employee abstract role. This is a one-time-only setup.

To create the provisioning rule for employees:

1. Navigate to the **Setup and Maintenance** work area.

2. On the **All Tasks** tab search for the **Manage HCM Role Provisioning Rules** task.

3. Click the **Go to Task** button for the **Manage HCM Role Provisioning Rules** task.

   The **Manage HCM Role Provisioning** page appears.

4. Click the **Create** button.

5. In the **Mapping Name** field enter the rule name, for example, Employee.

6. In the **Conditions** region, select **Employee** from the **Assignment Type** field.

7. Select **Active** for **Assignment Status**. This additional condition ensures that the role is automatically removed if the user is terminated.

8. In the **Associated Roles** region, use the **Add Row** button to add the Employee role.

9. Make sure the **Autoprovision** option is selected for the role.
10. Click **Save and Close**.

**Creating the Setup User**

To create the setup user:

1. On the **Navigator** select the **Manage Users** link under the **Manager Resources** heading.

   The **Manage Users** page appears.

2. Click the **Create** button.

   The **Create User** page appears.

3. In the **Personal Details** region, enter the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>Enter the user’s last name. Entry is required.</td>
</tr>
<tr>
<td>First Name</td>
<td>Optionally, enter the user’s first name.</td>
</tr>
<tr>
<td>E-Mail</td>
<td>Enter a unique e-mail address. This e-mail address is used to send the initial notification to the user and can be changed later.</td>
</tr>
</tbody>
</table>

4. In the **User Details** region, enter the user name. If you leave the **User Name** field blank, the application creates a user name based on the entries you already made.

5. If the **User Notification** region appears and the **Send User Name and Password** is unselected, then select it if you want to send the e-mail notification with the login and password to the user when you save the record.

   If this region is not present in the page, then the notification will be sent automatically.

6. In the **Employment Information** region, enter the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Type</td>
<td>Select <strong>Employee</strong>. Do not select <strong>Contingent Worker</strong> because enterprise role provisioning is based on the employee’s job, a field that appears only for employees.</td>
</tr>
</tbody>
</table>
Legal Employer

Select the legal employer created as part of enterprise setup.

If you are implementing in the Oracle Sales and Marketing Cloud Service, then the legal employer was set up for you based on the information that you provided when you signed up for the service.

Business Unit

Select the business unit created as part of enterprise setup.

If you are implementing in the Oracle Sales and Marketing Cloud Service, then the business unit was set up for you based on the information that you provided when you signed up for the service.

Job

Select the job you created earlier.

7. Click the Autoprovisioning Roles button.
   The Roles region should now display the following roles:
   • Application Implementation Consultant
   • IT Security Manager
   • Employee

8. Click Save and Close.

Changing User Names: Explained

You can change user names for both new and existing users.

Changing User Names When Creating Users

When you create a user using the Create User interface, the User Name field may be blank. If you leave it blank, then the user name is generated in the enterprise default format when you save the user.
Alternatively, you can enter a user name on the Create User page, regardless of whether the field is blank. Any name that you enter replaces the default user name.

The user is notified of the user name only if user notifications are enabled and the user has not yet been notified of any user name and password.

Changing Existing User Names

You can change an existing user name on the Manage User Account and Edit User pages. The new name is sent automatically to Oracle Identity Management.
However, it becomes visible in Oracle Fusion Applications only after the user signs in for the first time using the new name.

When you change an existing user name:

- The password is unchanged.
- Roles provisioned to the user are unchanged.
- The user is not notified automatically of the new name.

You are recommended to send details of the new user name to the user.

**FAQs for Define Users for CRM**

**What happens when I autoprovion roles?**

When you autoprovion roles to a user, the user’s assignments are reviewed automatically against all current role mappings.

Roles for which the *Autoprovion* option is selected are:

- Provisioned to the user immediately, if the user is eligible for the role and does not currently have it
- Deprovisioned from the user immediately, if the user is no longer eligible for the role but currently has it

You are recommended to autoprovion roles to individual users if you know that new or changed role mappings exist for which those users are eligible. Otherwise, roles are not provisioned or deprovisioned automatically until the user’s assignments are next updated.

**Where do default user names come from?**

By default, user names are defined by Oracle Identity Management (OIM). The user name is typically the user’s first and last names, but this format can be changed in OIM.

The default format of user names can be specified for the enterprise in Oracle Fusion HCM. This setting overrides the OIM default format. Your enterprise may have selected one of the following values:

- Person number
- Party number
- Primary work e-mail

If the default user-name format for the enterprise is the person number, then primary e-mail is used instead for party users who have no person number.
Default user names may not appear for new users until the relevant value (for example, the person number) is available. The user-account request is not submitted to OIM until the value exists.

**What happens when I link a user account?**

The current person or party record is linked to the selected Oracle Identity Management (OIM) user account. When you click **Save**, the request is submitted to OIM and the account status is **Requested**. Once the account status is **Active**, the user can sign in using the account.

Any roles currently provisioned to the linked account do not appear in the Roles section of the page until the account status is **Active**. However, you can add roles before clicking **Save**.

The **Link User Account** action appears only for persons or party users whose records are not already linked to a user account. When you link a user account, the person or party is not notified automatically. You are recommended to notify the user when the account is linked.

**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 Enterprise Governance, Risk and Compliance (GRC) 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 GRC, an access point is the lowest level access for a particular application. In GRC, 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 entitlement changes prior to putting them into production, and adjust controls as needed.

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.
Segregation of Duties in the Security Reference Implementation: Explained

Segregation of duties (SOD) is a special case of function security enforcement. A segregation of duties conflict occurs when a single user is provisioned with a role or role hierarchy that authorizes transactions or operations resulting in the possibility of intentional or inadvertent fraud.

The predefined SOD policies result in duty separation with no inherent violations. For example, an SOD policy prevents a user from entitlement to create both payables invoices and payables payments.

However, the most common duties associated with some job and abstract roles could conflict with the predefined segregation of duties. A predefined role hierarchy or job or abstract role may include such common duties that are incompatible according to a segregation of duties policy. For example, the predefined Accounts Payable Supervisor job role includes the incompatible duties: Payables Invoice Creation Duty and Payables Payment Creation Duty.

Every single predefined duty role is free from an inherent segregation of duties violation. For example, no duty role violates the SOD policy that prevents a user from entitlement to both create payables invoices and payables payments.

Jobs in the reference implementation may contain violations against the implemented policies and require intervention depending on your risk tolerance, even if you define no additional jobs or SOD policies.

Provisioning enforces segregation of duties policies. For example, provisioning a role to a user that inherits a duty role with entitlement to create payables invoices enforces the segregation of duties policy applied to that duty role and ensures the user is not also entitled to create a payables payment. When a role inherits several duty rules that together introduce a conflict, the role is provisioned with a violation being raised in the Application Access Controls Governor (AACG). If two roles are provisioned to a user and introduce a segregation of duties violation, the violation is raised in AACG.

**Note**

SOD policies are not enforced at the time of role definition.

Aspects of segregation of duties policies in the security reference implementation involve the following.

- Application Access Controls Governor (AACG)
- Conflicts defined in segregation of duties policies
- Violations of the conflicts defined in segregation of duties policies

**Application Access Controls Governor (AACG)**

AACG is a component of the Oracle Enterprise Governance, Risk and Compliance (GRC) suite of products where segregation of duties policies are defined.
• 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.

Your risk tolerance determines how many duties to segregate. The greater the segregation, the greater the cost to the enterprise in complexity at implementation and during maintenance. Balance the cost of segregation with the reduction of risk based on your business needs.

Conflicts

An intra-role conflict occurs when a segregation of duties policy expresses constraints within the construct of a single role (entitlement and duties) that creates violations.

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.

Violations

A segregation of duties violation occurs when a policy is defined that allows a segregation of duties conflict to occur.

Notifications report conflicts to the requester of the transaction that raised the violation. Oracle Identity Management (OIM) shows the status of role requests indicating if a segregation of duties violation has occurred.

For information on configuring audit policies, see the Oracle Fusion Applications Administrator's Guide.

For more information on managing segregation of duties, see the Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User's Guide.

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 predefined in Oracle Fusion Applications and available when creating SOD policies.

- User Status
- User Name
- 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.

---

Aspects of segregation of duties policies in the security reference implementation involve the following.

• Application Access Controls Governor (AACG)
• Conflicts defined in segregation of duties policies
• Violations of the conflicts defined in segregation of duties policies

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 an Oracle Enterprise Governance, Risk and Compliance (GRC) 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 Manager 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 Manager 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 are not actually violations, or are violations 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 subjected 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 in Oracle Enterprise Governance, Risk and Compliance (GRC). 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 Enterprise Governance, Risk and Compliance (GRC) 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 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.

Creating Partner User Accounts: Explained

When you create a partner user, you enable the partner member to access and use the deploying company's resources for working on assigned tasks. You also assign job and security roles to the partner member's user profile and specify the organization to which the new user needs to belong.

Assigning Job and Security Roles to Partner Users

Every partner user needs to have an assigned job role. This job role can be used to create security roles for the user. Based on the security roles you assign, the user can access applications, locations, and data within the deploying company. You may choose to assign security roles automatically to a user; you can also assign additional security roles individually if needed.

Assigning Partner Users to Organizations

While assigning non-manager partner users to organizations, you can either select an organization or a manager. Once you select an organization, the manager of the selected organization becomes the new user's manager. Similarly, once you select a manager, the new user automatically becomes a member of the organization to which the manager belongs.

Assigning Manager-Level Partner Users to Organizations

If the role of the new user is that of a manager, you need to assign the new user to an organization even after specifying a manager. This is because you granted the new user a managerial role, and you now need to specify the organization that the new user needs to manage. You can either select an organization from the list of available organizations, or you can create a new one if required.
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 for Developers.

Overview

• Approval management is an extension of the human workflow services of Oracle SOA Suite. Refer to the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management.

  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

• Task configuration involves managing policies that control approval flows. Refer to the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management.

  See: Using Task Configuration

  • To configure a predefined approval policy, select the predefined rule set and click the Edit task icon button.

  • To disable a predefined rule set, select the Ignore this participant check box for that rule set.

  • To edit the rules within a predefined rule set, you can insert, update, or delete from the seeded rules as needed while in edit mode.
• You can configure a specific rule to automatically approve without being sent to any approver. Modify the routing for that rule so that it is sent to the initiator (which means the requestor is the approver), set the Auto Action Enabled option to True, and enter APPROVE in the Auto Action field.

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 for Developers.

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. 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. Help that is embedded in the application, for example hints, can also be customized.

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.

Local Installation of Help

Select the Local Installation of Help feature choice so that 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 access to features that involve navigation to sites on the Web. If you select this feature choice, then the Web
Sites Available from Help Site section is available on the Set Help Options page. For Oracle Cloud, always leave this feature choice selected so that your users can access the Cloud Learning Center.

**Help Customization**

Select the Help Customization feature choice if you intend to customize predefined help or add your own files to help. For example, you can add internal policies or procedures as help, and Oracle User Productivity Kit content, if any. Only users with job roles containing the Application Help Text Administration duty role have access to customize help.

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
- Privacy Statement

**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. The collaboration features are also used elsewhere in Oracle Fusion Applications. Discussions may not be available in Oracle Cloud implementations.

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 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 users with the Application Help Text Administration duty role have access to the Manage Announcements icon button in the Announcements sections. They can create, edit, and delete announcements for the tab that they are on, and set the date range for when each announcement is to be displayed.

Note

Use the full URL, for example http://www.oracle.com, when creating links.

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.

You can set a help option to enable discussions. Each help file would contain a Discuss link that all users 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 in your enterprise.

Important

Do not enable discussions until servers for discussions are up and running.

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.

The scope of ratings and reviews is limited to your enterprise.

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>
</tbody>
</table>
Is there a specific time period for which this access is needed? | No, the help files should always be viewed only by the HR specialists
---|---
Where do you want this group to appear in the list of values for help security groups? | First

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>
<tr>
<td>Description</td>
<td>Viewing by HR specialists only</td>
</tr>
<tr>
<td>Display Sequence</td>
<td>1</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_HUMAN_RESOURCE_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.

**FAQs for Define Help Configuration**

**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 Application Help Text Administration duty role have access to customize help in Oracle Fusion Applications Help. This duty is assigned by default to various job roles, in particular the administrators for product families.

You can assign the duty role to other users who need access to customize help. Use the Manage Duties task in the Setup and Maintenance work area to search for the Application Help Text Administration duty role on the Role Catalog page, and map additional job roles to this duty role.

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.

Note

The Define Application Toolkit Configuration task list is available in implementation projects only if the Application Toolkit Component Maintenance feature choice is selected.
Use the Setup and Maintenance work area to access the tasks in the Define Application Toolkit Configuration task list.

**Map Reports to Work Areas**

**Additional Report Setup in the Context of the Reports and Analytics Pane:**

**Highlights**

Aside from determining which work areas a specific report is mapped to, you can perform additional setup for reports in the context of the Reports and Analytics pane. You can set up report permissions, and enable Oracle Business Intelligence (BI) Publisher reports for scheduled submission.

This additional setup is described in the Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition and the Oracle Fusion Applications Extensibility Guide for Business Analysts.

**Report Permissions**

- You can restrict access to specific reports for specific users, and this security is not limited to the Reports and Analytics pane. Refer to the Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition.

  See: Assigning Permissions

**Oracle Business Intelligence Publisher Reports Submission**

- Oracle BI Publisher reports must be registered as processes with Oracle Enterprise Scheduler to be enabled for scheduling. This registration also enables a Schedule link for the report in the Reports and Analytics Pane. Refer to the Oracle Fusion Applications Extensibility Guide for Business Analysts, and perform the following steps in the specified order.

  - Create an Oracle Enterprise Scheduler job definition for the report.
  - Specify the job definition details in the report's properties.

**FAQs for Map Reports to Work Areas**

**How can I set up the Reports and Analytics pane for all users?**

You can remove any currently mapped report from the Reports and Analytics pane, or add mappings to reports from the Oracle Business Intelligence (BI) Presentation catalog. To access the setup, click **Edit Settings** in the Reports and Analytics pane, or use the Map Reports to Work Areas task in the Setup and Maintenance work area. If you do the former, then you can set up only the Reports and Analytics pane on the work area that you are in.
If you do the latter, then you can select a work area to set up. If you do not see the desired work area, most likely you do not have access to it due to security. You can request to be granted a role that has access to the work area, or another administrator or business user with access to the work area can be granted the Reports and Analytics Region Administration Duty to be able to map reports to the work area.

Tip
On the Map Reports to Work Areas page only, you can also use the Synchronize button to remove mappings to reports that are no longer in the catalog, for all work areas at once.

Any changes you make in either UI apply to all users with access to the mapped work area.

Why can't I see reports when mapping reports to work areas for the Reports and Analytics pane?

It is possible that there are no reports currently mapped to the work area that you select in the Map Reports to Work Areas page. Alternatively, reports are mapped, but you do not see 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.

Why can't I see reports when I edit settings for the Reports and Analytics pane?

In the Edit Settings window, you may not be able to see a currently mapped report because you do not have access to it 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.

Set Watchlist Options

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
Define Common Applications Configuration for Marketing

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.

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.
Maintain Common Reference Objects

Maintain Common Reference Objects: Overview

The Maintain Common Reference Objects task list contains Oracle Middleware Extensions for Applications (Applications Core) tasks that support 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. You can access this task list by starting in the Setup and Maintenance Overview page and searching for common reference object task lists.

For more information on configuring custom objects, see the Oracle Sales Cloud Extensibility Guide.

To make the Maintain Common Reference Objects task list available in your implementation project, go to Setup and Maintenance Overview - Configure Offerings, and for a specific offering, select the Maintain Common Reference Objects feature choice.

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.
  
  See: Characteristics of the Level Categories
  
  See: Benefits of a Logical Hierarchy

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 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...
Maintain Common Reference Objects

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.

Define ISO Reference Data

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.

**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).

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 ledgers, 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.

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.

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 Portal: 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

Menu Customization: Explained

You use the Manage Menu Customizations task to customize the navigator and home page menus. This task is available from the Setup and Maintenance work area, which is accessible from the Administration menu in the Oracle Fusion Applications global area. Select either Customize - Navigator or Customize - Homepage to proceed with the customization activity.

Note
To perform menu customization at runtime, it is important that you have the required privileges.

You customize the menus at the site level and your changes affect all users (or all users of a tenant if in a multi-tenant environment).

Tip

If you are making minor changes, such as adding or editing one or two nodes, then you can hide the changes until you have completed your customizations. However, if you are making more than minor changes, such as rearranging several nodes, you might want to instead create a sandbox before customizing menus.

Navigator Menu Configuration

The navigator menu is the global menu that is accessible from the Oracle Fusion Applications global area. It allows users to navigate directly to the pages inside Oracle Fusion Applications as well as to outside web pages. The menu is composed of links (items) that are organized in a hierarchy of groups.

You can customize the navigator menu to address needs that are specific to your organization. For example, you might want to add specialized groupings for cross-functional teams or add links to web pages or external applications. You can add groups and links to the navigator menu, as well as hide and show them.

The Manage Menu Customizations task displays the menu groups as expandable nodes, with which you can traverse the menu hierarchy.

Note

Not all Oracle Fusion Applications pages appear in the navigator menu, because some pages are accessible from a work area or from other links in the global area such as the Home link.

The following table lists the Navigator menu customization tasks that you can perform at runtime as well as the tasks that you cannot perform.

<table>
<thead>
<tr>
<th>Permitted Tasks</th>
<th>Restricted Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Add and delete custom groups.</td>
<td>• You cannot add menu items (links) as top-level nodes. You can add nodes to only the groups in the top level and subgroups.</td>
</tr>
<tr>
<td>• Edit any group.</td>
<td>• You cannot delete nodes that are delivered with the product. Instead, you can hide them.</td>
</tr>
<tr>
<td>• Add and delete custom items.</td>
<td>• You cannot move nodes. Instead, you must duplicate the node and hide the original node.</td>
</tr>
<tr>
<td>• Edit any item.</td>
<td></td>
</tr>
<tr>
<td>• Specify navigation for an item:</td>
<td></td>
</tr>
<tr>
<td>• Specify navigation to a UI Shell page in an Oracle Fusion application.</td>
<td></td>
</tr>
<tr>
<td>• Specify navigation to an external web page.</td>
<td></td>
</tr>
<tr>
<td>• Hide or show groups and items.</td>
<td></td>
</tr>
</tbody>
</table>
Home Page Menu Configuration

The home page menu is the set of tabs that are displayed in the Oracle Fusion Applications global area. The home page menu displays tabs for all the items in the menu for which the end user has access privileges. You can add tabs to the home page menu, as well as hide and show them.

The following table lists the Home page menu customization tasks that you can perform at runtime as well as the tasks that you cannot perform.

<table>
<thead>
<tr>
<th>Permitted Tasks</th>
<th>Restricted Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Add and delete custom items.</td>
<td>• You cannot add menu items (links) as sub-nodes. All nodes are top-level nodes.</td>
</tr>
<tr>
<td>• Edit any item.</td>
<td>• You cannot delete nodes that are delivered with the product. Instead, you can hide them.</td>
</tr>
<tr>
<td>• Specify navigation to a UI Shell page in an Oracle Fusion application.</td>
<td>• You cannot move nodes. Instead, you must duplicate the node and hide the original node.</td>
</tr>
<tr>
<td>• Hide or show items.</td>
<td></td>
</tr>
</tbody>
</table>

Adding Navigator Menu Group: Points to Consider

You arrange the navigator menu by building a hierarchy of nested groups.

Use the View menu to expand or collapse a group of nodes. You can also right-click a node and access similar actions to facilitate tree navigation.

Adding Groups

To add a group, you can insert a group above or below a peer group or insert a child group. You edit a group by defining a label and specifying whether the group should be rendered. You typically hide the group until all changes have been completed.

Adding Menu Items: Points to Consider

The home page menu items are URL links to home pages in Oracle Fusion applications. The Navigator menu items can either be links to UI Shell pages or links to external applications and web sites.

In the menu hierarchy, the home page menu items are always top-level items. Whereas, you can add Navigator menu items to top-level groups and to their subgroups but you cannot add navigator menu items as top-level nodes.

Adding a Home Page Menu Item

To add a home page menu item, navigate to the place where you want the item to appear and insert it above or below the existing item. You can also duplicate
 Maintain Common Reference Objects

Adding a Navigator Menu Item

To add a Navigator menu item, you navigate to the item’s group and insert the item above or below another item.

You can also duplicate an existing item. You must provide a label for the menu and either link the menu item to a UI Shell page or link it to an external web site or application.

You can link a Navigator menu item to the following:

- A UI Shell page in an Oracle Fusion application.
- A dynamic URL of a page outside of Oracle Fusion Applications where the host, port, or context root might change.
- A Static URL of a page outside of Oracle Fusion Applications where the host, port, or context root does not change.

Linking to a UI Shell Page

If the new item points to a UI Shell page in an application, then you must provide the name of the web application and the view ID of the target page. The web application name and view ID can be obtained from an existing menu item that links to the same UI Shell page.

In a non-Cloud implementation, you also can obtain the web application name from the context root for the application, and you can obtain the view ID from the id attribute for the page’s <view> tag in the product’s public_html/WEB-INF/afdc-config.xml file.

If you want secure access to the target UI Shell page from the menu item, then you must provide the name of the secured resource and the name of the policy store’s application stripe. When an end user clicks the link, the Oracle Fusion Applications checks the secured resource and the Lightweight Directory Access Protocol (LDAP) policy store to determine whether the user has the privilege to view the page.

If there is another menu item that points to the same page, then you can get the secured resource name and application stripe from that item. In a non-Cloud implementation, you also can obtain the application stripe from the jps.policystore.applicationid parameter in the application’s weblogic-application.xml file. Examples of application stripes are crm, fscm, and hcm.

For non-Cloud applications, you can determine the secured resource name by obtaining the name of the web page’s page definition file. By default, the page definition files are located in the view/PageDefs package in the Application Sources directory of the view project. If the corresponding JavaServer Faces (JSF) page is saved to a directory other than the default (public_html), or to a subdirectory of the default, then the page definition will also be saved to a package of the same name. An example of a secured resource name is oracle.apps.view.pageDefs.CaseList_Form_Attach_UIShellPagePageDef.
A UI Shell page might take parameters and display or act differently based on the parameters that are passed in. For example, if accessing a page from one group in the menu hierarchy, the parameter might be set to `status=Open` and if accessing the page from a different group, the parameter might be set to `status=Closed`. If the page takes parameters, you can use the Page Parameters List text box to provide a semicolon-delimited string of name-value pairs, such as `org=m1;context=s1`. You can use expression language (EL) to specify the parameters. If the EL evaluates to an Object, the `toString` value of that Object is passed as the value of the parameter.

**Linking to the Dynamic URL of an External Web Site or Application**

Linking a menu item to a dynamic URL is beneficial in cases where the host, port, or context root to which you point frequently changes. Instead of updating the link to each application, you can update the details of the web application in the topology registration, and that change affects all menu items that contain dynamic links pointing to that web application. For example, you would need a dynamic URL to link to a test version of an application and you will need to change the host and port when you move the application from a test environment to a production environment.

To link to a page outside of Oracle Fusion Applications where the host, port, or context root might change, you must first register the web application in the topology using the Register Enterprise Applications task.

While creating a new menu item on the Create Item Node dialog box, select the Dynamic URL option and provide the details of the web application as per the following example.

When the complete URL to be linked is: `http://example:9011/myApp/faces/Page1`,

- The name of the web application added to topology would be: `myApp` (the value that would eventually appear in the Web Application list) and the protocol host, port, and context root values of the URL would be: `http://example:9011/myApp`
- The value to be provided in the Destination for Web Application field would be: `/faces/Page1`

Once the menu item is linked to the dynamic URL, the target page appears in a new browser window or tab when you click the menu item.

**Linking to a Static URL of an External Web Site or Application**

This option is used when you link a menu item to a page outside Oracle Fusion Applications where the host, port, or context root remains constant. For example, you can use a static URL to link to `http://www.oracle.com`.

**Hiding or Displaying Menu Nodes: Points to Consider**

While you are creating or working with a menu group or a menu item, you might want to prevent end users from accessing the node. You can hide the
menu group or menu item while you are working with it, and then show the node when you have completed the task.

**Working with Nodes**

The Manage Menu Customizations page shows all nodes. The Rendered check box is selected by default for all nodes that are added and are visible.

To hide a node, clear the Rendered check box. You can edit the node anytime to either display or hide it.

If you want a menu group or a menu item to appear only if certain conditions are met, you can use an expression language (EL) command to make the node to appear. For example, `${securityContext.userInRole('ADMIN')}`.

A node that appears in italics either contains an EL command or the Rendered check box beside it was cleared, and therefore is hidden from end users.

---

**Tip**

For major changes that need to be tested and approved, you might want to use the sandbox manager instead of hiding and showing nodes.

---

**Design Time Menu Customizations: Highlights**

The menu customization feature provides several options to add, modify, and organize the Navigator and home page menus during design time. You must have developer rights to perform these customizations.

**Note**

Design time menu customizations are not applicable to Oracle Cloud implementations.

An overview of customizing the Navigator menu and home page is provided in the Oracle Fusion Applications Extensibility Guide.

**Customizations**

- Use Oracle JDeveloper to customize the Navigator and home page menus at design time.
  
  See: Customizing Menus

- Define translations for your customizations in the locales you support.

  See: Translating Menu Customizations

- Customize the page template to display the Navigator menu groups as separate menus, each of them displaying their list of menu items. Refer to the Oracle Fusion Applications Developer's Guide.

  See: Rendering the Navigator Menu as Dropdown Buttons
Troubleshooting Navigator Menu: Highlights

If the Navigator menu does not display customizations as expected, use the following troubleshooting tips to verify the changes.

Issues and Resolutions

• If an expected menu item does not appear in the Navigator menu, verify whether the menu item has been hidden from view.

• If a custom menu item was added and the browser does not display the page indicated by the URL, open the Manage Menu Customizations task and verify whether the web application name is the same as the context root for the application, and that the view ID is the id attribute for the page's <view> tag in the product's public_html/WEB-INF/adfc-config.xml file. The URL should not contain the .JSPX suffix.

• If you see a "webApp value not define" error message when you choose an item in the Navigator menu, verify whether the application is in the topology tables. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Viewing the Routing Topology of an Oracle Fusion Applications Instance, Product Family, or Product

Manage Audit Policies

Managing Audit Policies: Explained

Auditing is used to monitor user activity and all configuration, security, and data changes that have been made to an application. Auditing involves recording and retrieving information pertaining to the creation, modification, and removal of business objects. All actions performed on the business objects and the modified values are also recorded. The audit information is stored without any intervention of the user or any explicit user action.

Use audit policies to select specific business objects and attributes to be audited. The decision to create policies usually depends on the type of information to be audited and to the level of detail that is required to be reported.

Enabling Audit Functionality

To enable audit, ensure that you have administrative privileges. For Oracle Fusion Applications, you must configure the business objects and select the attributes before enabling audit. If you enable audit without configuring the business objects, auditing remains inactive. By default, auditing is disabled for all applications.

To enable auditing for Oracle Fusion Middleware products, select one of the levels at which auditing is required for that product. The audit levels are...
predefined and contain the metadata and events to be audited. For more
information, refer to the Oracle Fusion Middleware documentation and also
the Oracle Enterprise Repository for Oracle Fusion Applications at http://
fusionappsoer.oracle.com.

If you do not want an application to be audited, you can stop the audit process
by setting the Audit Level option to None. While viewing the audit report for
that application, you can specify the period during which auditing remained
enabled.

**Configuring Audit Business Object Attributes: Points to Consider**

Audit allows you to track the change history of particular attributes of a business
object. However, those objects and their attributes must be selected for audit
and auditing must be enabled for that application. Your configuration settings
determine which attributes to audit for a given object, and when the audit starts
and ends. Auditing takes into account all the create or insert, update, and delete
operations performed on an object and its attributes.

To configure audit business object attributes, navigate to the Manage Audit
Policies page in the Setup and Maintenance work area.

**Selecting an Application**

To set up auditing, you must select a web application that contains the required
business objects that can be audited. From the list of business objects, select those
business object that you want to audit. Selecting a business object also displays
its attributes that are enabled for auditing.

**Selecting Attributes**

For each selected business object to be audited, select the corresponding
attributes to include in the audit. All attributes that belong to that object are by
default selected for audit and appear on the user interface. However, you can
add or remove attributes from the list. When you remove an attribute from the
list, you stop auditing it even when the parent object is selected for audit. So, if
you want an attribute to be audited, you must add it to the list.

**Note**

If the object selected in an audit hierarchy is also a part of several other audit
hierarchies, the attribute configuration for that object is applicable to all the
hierarchies in that application.

**Starting and Stopping Audit**

The business object is ready for audit after you select its attributes and save the
configuration changes. However, to start auditing, the audit level for Oracle
Fusion Applications must be set to Auditing on the Manage Audit Policies page.

To stop auditing an object, you can deselect the entire object and save the
configuration. As a result, all its selected attributes are automatically deselected.
and are not audited. To continue to audit the business object with select attributes, deselect those attributes that are not to be audited.

When end-users view the audit history for an application, they can specify the period for which they want the results. Therefore, it is important to note when you start and stop auditing an application. For example, today if end-users intend to view the audit history of an object for the previous week, but auditing for that object was stopped last month, they would not get any audit results for that week because during the entire month that object was not audited. Even if you enable audit for that object today, end-users cannot get the wanted results because audit data until today is not available.

**Configuring Audit: Highlights**

You can set up auditing for Oracle Fusion Applications using the Manage Audit Policies page in the Setup and Maintenance work area of Oracle Fusion Applications.

To set up auditing for Oracle Fusion Middleware products, you must select the level of auditing that maps to a predefined set of metadata and events that have to be audited. Information on configuring audit for Oracle Fusion Middleware products is provided in Oracle Fusion Middleware guides.

You can also create a configuration file and deploy it to audit a specific Oracle Fusion Middleware product. The configuration details for Oracle Fusion Middleware products are available in the form of audit-specific assets that can be used to create the configuration file (config.xml). For more information, see the Oracle Enterprise Repository for Oracle Fusion Applications at http://fusionappsoer.oracle.com, and search with Audit as the Asset Type to get the list of audit-specific assets.

**Oracle Fusion Middleware Products**

  
  See: Auditing Web Services

**Oracle Fusion Security Products**

- Configure business objects to enable auditing in Oracle Fusion security products. Refer to Oracle Fusion Middleware Application Security Guide.
  
  See: Oracle Fusion Middleware Audit Framework Reference

**Manage Oracle Social Network Objects**

**Managing Oracle Social Network Objects: Explained**

Use the **Manage Oracle Social Network Objects** task for managing the Oracle Social Network Objects. The integration of Oracle Social Network Cloud
Service with applications and business processes brings key attributes from the applications to share, socialize, and update information. This helps in making better business decisions based on additional information that you obtain and analyze within your social network environment.

Use the Manage Oracle Social Network Objects page to set up and define:

- The business objects and attributes to enable
- The enablement method for social network integration with Oracle Fusion Applications

You can access the Manage Oracle Social Network Objects page by starting in the Setup and Maintenance Overview page and searching for the task named Manage Oracle Social Network Objects.

Use social network to:

- Discuss projects and plans in public forums
- Maintain:
  - Membership groups
  - Activity feeds of the people you choose
- Facilitate:
  - One-on-one Conversations
  - Reviews
  - Document sharing

**Note**

Oracle Social Network Cloud Service is currently available in Cloud implementations only.

An important aspect of managing Oracle Social Network objects is enabling business objects for integration.

**Enabling Business Objects for Integration**

A business object can't be shared within social network until a functional administrator or implementor:

- Accesses the Manage Oracle Social Network Objects page in Oracle Fusion Applications
- Enables the business object for social network integration

**Options for Enabling Oracle Social Network Objects: Explained**

To enable business objects and apply attributes for Oracle Social Network Cloud Service integration with Oracle Fusion Applications, use the Manage Oracle Social Network Objects task.
In the **Manage Oracle Social Network Objects** page, you can:

- Enable an object
- Disable an object
- Enable all objects
- Enable business object attributes

To access the **Manage Oracle Social Network Objects** page:

1. Search for the **Manage Oracle Social Network Objects** task in the Setup and Maintenance work area.
2. In the Search Results section, click the **Go to Task** icon to open the **Manage Oracle Social Network Objects** page.

---

**Note**

Custom objects and attributes created in Application Composer in the mainline are also displayed on the **Manage Oracle Social Network Objects** page. You can enable these objects and attributes for social network integration.

---

### Enable Object

To enable a business object:

1. Access the **Manage Oracle Social Network Objects** page.
2. In the **Business Objects** section, select a business object, click **Enable Object**, and select one of the enablement options. The available enablement options are:

   - **Manual**: (Recommended) Empowers the user to decide whether to share each instance of the object with social network.

   **Note**

   Once shared, all updates to the enabled attributes of the object instance, and deletes, are sent to social network. Updates to attributes that are not enabled are not sent.

   - **Automatic**: Automatically sends the newly enabled object instances and updates to social network.

   **Note**

   All object instances are automatically shared with social network upon creation, and all subsequent updates to the enabled attributes of the object instances, and deletes, are automatically sent to social network.

   - **No**: Does not send any information on object instance to social network. This is the default option.

3. Click **OK**.

---
This enables the selected business object, and empowers the user to decide whether to share each instance of the object with social network.

**Note**

After you enable an object, you must enable one or more attributes in the *Attributes* section of the *Manage Oracle Social Network Objects* page. Updates to enabled attributes are sent to social network.

**Disable Object**

To disable a business object:

1. Access the *Manage Oracle Social Network Objects* page.
2. In the *Business Objects* section, select a business object, and click *Disable Object*.
3. Save your changes.

This disables the selected business object by updating the enablement option as *No*.

**Enable All**

To enable all business objects:

1. Access the *Manage Oracle Social Network Objects* page.
2. In the *Business Objects* section, click *Enable All*.
3. Save your changes.

This enables all business objects in bulk, and updates the enablement option of all business objects as *Manual*.

**Note**

- After you enable business objects, you must enable one or more attributes in the *Attributes* section of the *Manage Oracle Social Network Objects* page. Updates to the enabled attributes are sent to social network.
- If you enable a business object, but don’t configure any attributes for the enabled business object, no attributes are sent to social network during create and update. The only exception is that some internal bookkeeping information are sent. Deletes are sent as usual.

**Status Column**

The Status column in the *Business Objects* table visually indicates:

- Whether a business object is enabled
- Which enabled business objects don’t yet have an enabled attribute assigned

The status indicators include:

- A check mark, which indicates that you have configured attributes for an enabled business object
A warning sign, which indicates that you have not configured any attributes for an enabled business object

**Enable Business Object Attributes**

To enable business object attributes:

1. In the Attributes section, click **Add** to display the **Select Attributes** dialog where you can select attributes to add to the table.
2. Select an attribute name in the table, and select the **Enabled** check box to enable the attribute.
3. Click **OK**.
4. Save your changes.

In the Attributes table, you can also:

- Click **View** to view a list of all attributes that are enabled.
- Click **Remove** to remove attributes from the table.
- Hover over the **Attribute Information** icon displayed next to descriptive flexfield attributes to view information about the attributes.

**Update Translations: Explained**

The Update Translations process sends attribute labels and business object names to Oracle Social Network Cloud Service for use in the user interface.

In social network, the attribute or business object labels appear in the language of your locale. If you change the locale in social network, then the attribute or business object labels appear in the updated language. However, the data appears in the language in which it was originally sent to social network. If you have previously sent an instance of the business object to social network, then the instance data isn’t updated. Clicking the **Update Translations** button on the **Manage Oracle Social Network Objects** page sends translations for business objects with the enablement option as **Manual** or **Automatic**.

**Synchronize Business Objects: Explained**

Use the **Synchronize** button on the **Manage Oracle Social Network Objects** page to synchronize business objects. This re-sends the definitions of business objects having the enablement option as **Manual** or **Automatic** to Oracle Social Network Cloud Service.

Use the Synchronize button at the:

- **Business Objects table level**: To re-send the definitions of a selected business object to social network. This button is enabled only when you select a row for a business object with the enablement option as **Manual** or **Automatic**.
• **Manage Oracle Social Network Objects page level**: To re-send the definitions of all business objects with the enablement option as Manual or Automatic to social network.

**Note**

If you had modified any business object enabled for social network and not saved your changes, then on clicking the **Synchronize** button, a warning message appears. This message informs you that you have not saved your changes, and you can select one of the following options:

- **Save and Synchronize**: To save the modified business objects, and synchronize the unmodified business objects.
- **Synchronize**: To ignore any unsaved business objects, and only synchronize the unmodified business objects.
- **Cancel**: To cancel the synchronization task.

**FAQs for Manage Oracle Social Network Objects**

**What happens if I update translations?**

When you update translations, you send translations for business objects with the enablement option as Manual or Automatic to Oracle Social Network Cloud Service.

On updating translations, you also:

- Synchronize the newly translated text from Oracle Fusion Applications so that it can be used within social network. This means you can:
  - Install and enable a new language.
  - Take a language patch at any time.
  - Send attribute labels and business object names to social network for use in its user interface.

**How can I update translations?**

Use the **Update Translations** button on the Manage Oracle Social Network Objects page for subsequent updates to labels and attributes.

Use the **Update Translations** button at the:

- **Business Objects table level**: To send translations for a selected business object to Oracle Social Network Cloud Service. This button is enabled only when you select a row for a business object with the enablement option as Manual or Automatic.
- **Manage Oracle Social Network Objects page level**: To send translations for all business objects with the enablement option as Manual or Automatic to social network.
**Note**

When you save the enablement of a business object to social network, it sends the translations as well. Hence, you need not click the *Update Translations* button after saving the enablement.

---

**When do I update translations?**

Run the *Update Translations* process only after you install a new language pack of Oracle Fusion Applications.

Updating translations synchronizes the newly translated text to Oracle Social Network Cloud Service for integration with Oracle Fusion Applications.

**Note**

When you save the enablement of a business object to social network, it sends the translations as well. Hence, you need not click the *Update Translations* button after saving the enablement.

---

**What happens if I synchronize business objects?**

When you synchronize business objects, you re-send the definitions of business objects having the enablement option as *Manual* or *Automatic* to Oracle Social Network Cloud Service.

**When do I synchronize business objects?**

Run the Synchronize process after the import of Customization Set Migration (CSM).

You can also run the process any time you want to synchronize the settings of business objects with Oracle Social Network Cloud Service without making any changes in the *Manage Oracle Social Network objects* page.

---

**Manage Applications Core Common Reference Objects**

**Manage Applications Core Messages**

**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.

**Manage Applications Core Administrator Profile Values**

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. To create or edit a message, navigate to the Manage Messages page in the Setup and Maintenance work area.

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 comprises various components, some of which are displayed only to select users. To determine which component of the message text is displayed to a particular user, set the Message Mode profile option (FND_MESSAGE_MODE) at the user level for that user. The message component short text is visible to all users and therefore, the profile option does not apply to this component. Also, the profile option applies only to messages in the message dictionary.

See: About Message Components

- Tokens are variables that represent values to be displayed in the message text.

See: About Tokens

**Profile Options and Related General Preferences: How They Work Together**

Some Oracle Middleware Extensions for Applications profile options are related to general preferences in the global area.

**Preferences**

The related general preferences are Default Application Language, Territory, Date Format, Time Format, Currency, and Time Zone. When the user changes any of these preferences, the stored values in LDAP are updated accordingly.

**Profile Options**

The corresponding profile options are Default Language, Default Territory, Default Date Format, Default Time Format, Default Currency, and Default User Time Zone. No matter what you set for these profile options at any level, the preferences settings, or LDAP values, take precedence. The profile option value is used only if the LDAP value is not available. Updating the profile option value does not automatically update the value in LDAP or preferences.

**FAQs for Manage Applications Core Administrator Profile Values**

**How can I enable the privacy statement?**

Use the Privacy Statement URL profile option to enable the Privacy Statement menu item in the global area. This menu item in the Settings and Actions menu is disabled by default.

Open the Setup and Maintenance work area, and use the Manage Applications Core Administrator Profile Values task to find the Privacy Statement URL profile.
option. In the Profile Value column, enter the full URL of the web page that contains the privacy content you want the menu item to link to.
Define Applications Core Configuration

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.

Once the customization level is set for a lookup type, it cannot be modified. The customization level for lookup types created using the Define Lookups page is by default set at the User level.

Standard, Common, and Set-Enabled Lookups

The available kinds of lookups are as follows.
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.

For more information on the predefined lookups and lookup codes, see assets with the Lookup type in the Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

### 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 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 lookups 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 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>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.

The message dictionary is described in the Oracle Fusion Applications Developer’s Guide.

**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, such as confirmations and field validations, are managed either in the Oracle Application Development Framework or through message resource bundles used for translation.

**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. To create or edit a message, navigate to the Manage Messages page in the Setup and Maintenance work area.

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 comprises various components, some of which are displayed only to select users. To determine which component of the message text is displayed to a particular user, set the Message Mode profile option (FND_MESSAGE_MODE) at the user level for that user. The message component short text is visible to all users and therefore, the profile option does not apply to this component. Also, the profile option applies only to messages in the message dictionary.

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_CNM, but use another suitable convention. The message number should be within the range that is designated for the product.

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
Define Applications Core Configuration

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.

**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.

Auditing Document Sequences

You can audit document sequences, if required, to provide an audit trail of the document sequences used in a specific product. However, before enabling the audit functionality for a document sequence, you must have created an audit table for the specific document sequence, using appropriate details. Enabling the audit functionality is permitted only for newly created document sequences. You cannot audit document sequences that are already in use by a specific product.

For more information about defining a document sequence audit table, see the Oracle Fusion Applications Developer’s Guide.

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.
Define Applications Core Configuration

- 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>Status</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</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>

**Tree Structure Audit Results: Explained**

Use the tree structure audit results to verify the tree structure’s correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- The result of the validation, including a detailed message
- Corrective actions to take if there are any validation errors

**Running an Audit**

Setting the status of a tree structure to active automatically triggers an audit of that tree structure. You can also manually trigger an audit on the manage Tree Structures page, using **Actions - Audit**. The Tree Structure Audit Result table shows a list of validations that ran against the selected tree structure.

**Validation Details**

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

<table>
<thead>
<tr>
<th>Validator</th>
<th>Description (what is checked)</th>
<th>Possible Cause for Validation Failure</th>
<th>Suggested Corrective Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Restrict By Set ID</strong></td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Set ID check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have a reference data set attribute. This validation does not take place when the check box is not selected.</td>
<td>Even when the check box is selected, one or more of its data source view objects does not contain a reference data set attribute.</td>
<td>If reference data set restriction is required for this tree structure, include a reference data set attribute on all data sources. Otherwise, deselect the check box.</td>
</tr>
<tr>
<td><strong>Row Flattened Table Name</strong></td>
<td>On the Manage Tree Structures: Specify Performance Options page, a valid row flattened table must be specified for the tree structure. It can either be the standard row flattened table <code>FND_TREE_NODE_RF</code> or a custom table.</td>
<td>• The specified table does not exist in the database. • The specified table does not contain the same columns as the <code>FND_TREE_NODE_RF</code> table.</td>
<td>Correct the row flattened table definition.</td>
</tr>
</tbody>
</table>
| Available Label Data Sources | On the Manage Tree Structures: Specify Data Sources page, if a labeling scheme is specified for the tree structure by selecting a list item from the **Labeling Scheme** list box, the label data source view object specified for each data source must be accessible, and the primary keys must be valid. This restriction does not apply when you select **None** from the **Labeling Scheme** list box. | • Any of the specified label data source view objects do not exist.  
• Any of the specified label data source view objects do not have primary keys.  
• When a label data source view object is initially defined, the database registers the primary keys for the view object. If the view object is later modified such that its primary keys no longer match the primary keys that were registered earlier, this validation fails.  
• Correct the specified label data source view object.  
• Correct the primary keys of the specified label data source view object.  
• Either correct the primary keys in the label data source view object to match the primary keys that were earlier registered in **FND_TS_DATA_SOURCE**, or correct the primary keys registered in that table to match the new view object definition. |
Available Data Sources

Each data source view object specified for the tree structure must be accessible, and all its primary key attributes must be valid.

- Any of the specified data source view objects do not exist.
- When a data source view object is initially defined, the database automatically registers the primary keys for the view object if the **Use non-defined primary key columns** check box on the Data Source dialog box is not selected. If the check box is selected, the database registers the primary keys specified explicitly by the user on the Add Data Source dialog box. If the registered primary keys contain any duplicates, this validation fails.
  - The **Use non-defined primary key columns** check box is selected in a data source, but the list of specified primary key columns does not match the primary keys defined in the corresponding data source view object.
- Any common attribute that exists in both the data source view object and the tree node view object is not of the same data type in both view objects.
- Correct the specified data source view object.
- Correct the duplicate column in the registered primary keys.
- Correct the primary keys of the specified data source view object.
- Correct any mismatch in data types.
| Column Flattened Table Name | On the Manage Tree Structures: Specify Performance Options page, a valid column flattened table must be specified for the tree structure. It can either be the standard row flattened table FND_TREE_NODE_CF or a custom table. | • The specified table does not exist in the database.  
• The specified table does not contain the same columns as the FND_TREE_NODE_CF table. | Correct the column flattened table definition. |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Restrict by Date</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have effective start date and effective end date attributes. This validation does not take place when the check box is not selected.</td>
<td>Even when the check box is selected, one or more of its data source view objects does not contain effective start date and effective end date attributes.</td>
<td>If the date restriction is required for this tree structure, include the effective start date and effective end date attributes on all data sources. Otherwise, deselect the check box.</td>
</tr>
</tbody>
</table>
| Tree Node Table Name | On the Manage Tree Structures: Specify Definition page, a valid tree node table must be specified for the tree structure. It can either be the standard row flattened table FND_TREE_NODE or a custom table. | • No table is specified in the Tree Node Table field.  
• The specified table does not exist in the database.  
• The specified table does not contain the same columns as the FND_TREE_NODE table. | Correct the tree node table definition. |
| Allow Node Level Security | If the Allow Node Level Security option is set to No for the tree structure, the same option cannot be set to Yes on any of its data sources. This is a database setting that is not visible on the Manage Tree Structures page. | The option is set to No for the tree structure but one or more associated data sources have that option set to Yes. | Correct the option setting in the tree structure and their data sources. |

**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.

Tree Version Audit Results: Explained

Use the tree version audit results to verify the tree version’s correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- The result of the validation, including a detailed message
- Corrective actions to take if there are any validation errors

Running an Audit

An audit automatically runs whenever a tree version is set to active. You can also manually trigger an audit on the Manage Trees and Tree Versions page, using Actions - Audit. The Tree Version Audit Result table shows a list of validations that ran against the selected tree version.

Validation Details

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

<table>
<thead>
<tr>
<th>Validator</th>
<th>Description (what is checked)</th>
<th>Possible Cause for Validation Failure</th>
<th>Suggested Corrective Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effective Date</td>
<td>The effective start and end dates of the tree version must be valid.</td>
<td>The effective end date is set to a value that is not greater than the effective start date.</td>
<td>Modify the effective start and end dates such that the effective start date is earlier than the effective end date.</td>
</tr>
<tr>
<td>Root Node</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Allow Multiple Root Nodes check box for the Restrict Tree Node List of Values Based on option is not selected, and if the tree structure is not empty, the tree version must contain exactly one root node. This validation does not take place if the check box is selected.</td>
<td>Even if the check box is deselected, the tree version has multiple root nodes.</td>
<td>Modify the tree version such that there is exactly one root node.</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Data Source Max Depth</td>
<td>For each data source in the tree structure, on the Data Source dialog box, if the data source is depth-limited, the data in the tree version must adhere to the specified depth limit. This validation does not apply to data sources for which the Maximum Depth field is set to Unlimited.</td>
<td>The tree version has data at a depth greater than the specified depth limit on one or more data sources.</td>
<td>Modify the tree version such that all nodes are at a depth that complies with the data source depth limit.</td>
</tr>
<tr>
<td>Duplicate Node</td>
<td>On the Data Source dialog box, if the Allow Duplicates check box is not selected, the tree version should not contain more than one node with the same primary key from the data source. If the check box is selected, duplicate nodes are permitted.</td>
<td>Even when the check box is deselected, the tree version contains duplicate nodes.</td>
<td>Remove any duplicate nodes from the tree version.</td>
</tr>
</tbody>
</table>
| Available Node | All nodes in the tree version should be valid and available in the underlying data source. | • A node in the tree version does not exist in the data source. Deleting data items from the data source without removing the corresponding nodes from the tree version can result in orphaned nodes in the tree version. For example, if you added node A into your tree version, and subsequently deleted node A from the data source without removing it from the tree version, the validation fails.  
  
  • The tree version contains a tree reference node, which references another tree version that does not exist. | Remove any orphaned nodes from the tree version. Update tree reference nodes so that they reference existing tree versions. |
<p>| Node Relationship | All nodes must adhere to the relationships mandated by the data sources registered in the tree structure. | The tree structure has data sources arranged in a parent-child relationship, but the nodes in the tree do not adhere to the same parent-child relationship. For example, if the tree structure has a Project data source with a Task data source as its child, Task nodes should always be under Project nodes in the tree version. This validation fails if there are instances where a Project node is added as the child of a Task node. | Modify the tree version such that the nodes adhere to the same parent-child relationships as the data sources. |</p>
<table>
<thead>
<tr>
<th>SetID Restricted Node</th>
<th>On the Manage Tree Structures: Specify Data sources page, if the <strong>Set ID</strong> check box is selected to enable the <strong>Restrict Tree Node List of Values Based on</strong> option for each tree node, the underlying node in the data source must belong to the same reference data set as the tree itself. This restriction does not apply when the check box is not selected.</th>
<th>Even when the check box is selected, the tree version has nodes whose data source values belong to a different reference data set than the tree.</th>
<th>Modify the tree version such that all nodes in the tree have data sources with reference data set matching that of the tree.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label Enabled Node</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if a labeling scheme is specified for the tree structure by selecting a list item from the <strong>Labeling Scheme</strong> list box, all nodes should have labels. This restriction does not apply when you select <strong>None</strong> from the <strong>Labeling Scheme</strong> list box.</td>
<td>The tree structure has a labeling scheme but the tree version has nodes without labels.</td>
<td>Assign a label to any node that does not have a label.</td>
</tr>
<tr>
<td>Date Restricted Node</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each node in the underlying data source must have an effective date range same as the effective date range of the tree version. This restriction does not apply if the check box is not selected.</td>
<td>Even when the check box is selected, there are data source nodes that have a date range beyond the tree version’s effective date range. For example, if the tree version is effective from Jan-01-2012 to Dec-31-2012, all nodes in the tree version must be effective from Jan-01-2012 to Dec-31-2012 at a minimum. It is acceptable for the nodes to be effective for a date range that extends partly beyond the tree version’s effective date range (for example, the node data source value is effective from Dec-01-2011 to Mar-31-2013). It is not acceptable if the nodes are effective for none or only a part of the tree version’s effective date range (for example, the node data source value are effective only from Jan-01-2012 to June-30-2012).</td>
<td>Ensure that all nodes in the tree version have effective date range for the effective date range for the tree version.</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Multiple Active Tree Version</td>
<td>On the Manage Tree Structures: Specify Definition page, if the Allow Multiple Active Tree Versions check box is not selected for the tree structure, there should not be more than one active tree version under a tree at any time. This restriction does not apply if the check box is selected.</td>
<td>Even when the check box is not selected, there is more than one active tree version in the tree for the same date range.</td>
<td>Set no more than one tree version to Active within the same date range and set the others to inactive or draft status.</td>
</tr>
<tr>
<td>Range Based Node</td>
<td>On the Data Source dialog box, if the Allow Range Children check box is not selected, range-based nodes are not permitted from that data source. This restriction does not apply if the check box is selected.</td>
<td>Even when the check box is not selected, there are range-based nodes from a data source.</td>
<td>Ensure that any range nodes in your tree version are from a data source that allows range children.</td>
</tr>
</tbody>
</table>
### Terminal Node

| On the Data Source dialog box, if the Allow Use as Leaves check box is not selected, values from that data source cannot be added as leaves (terminal nodes) to the tree version. This restriction does not apply if the check box is selected. | Even when the check box is not selected, values from a data source are added as leaf nodes (terminal nodes). | Modify the tree version such that all terminal nodes are from data sources for which this check box is selected. |

### Usage Limit

| On the Data Source dialog box, if the Use All Values option is selected to set the Usage Limit for the data source, every value in the data source must appear as a node in the tree. This restriction does not apply if None option is selected. | Even if the Use All Values option is selected, there are values in the data source that are not in the tree version. | For each data source value that is not yet available, add nodes to the tree version. |

## 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
Define Applications Core Configuration

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.

<table>
<thead>
<tr>
<th>Type of Information</th>
<th>Profile Option Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>User preferences</td>
<td>Settings to provide access to social networking features</td>
</tr>
<tr>
<td>Installation information</td>
<td>Setting to identify the location of a portal</td>
</tr>
<tr>
<td>Configuration choices</td>
<td>Settings to change user interface skins and behaviors</td>
</tr>
<tr>
<td>Processing options</td>
<td>Settings to affect how much information to log either for an entire site or a specific user</td>
</tr>
</tbody>
</table>

You can add and configure new profile options in addition to configuring predefined profile options that are implemented as updatable.

**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.
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.

For more information on the predefined profile options, see assets with the Profile Option type in the Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

### 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.

<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.

```sql
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_LANUGUAGE_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 Customer Relationship Management (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>
</tbody>
</table>
Values at the site level take effect 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 following table demonstrates the FND_LANGUAGE profile option settings that would apply to specific users, based on the example above. For example, the user Hima is using the CRM Application Composer product, in the InFusion site. The example above shows that this profile option is set to Hindi at the user level for Hima. Because user is the highest applicable level for Hima, the applicable profile option value is Hindi for Hima.

<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
Define Flexfields

Flexfields: Overview

A flexfield is an extensible set of placeholder fields in application pages that are associated with a business object. Each segment of the flexfield corresponds to a single application field, such as a segment of a key identifying a particular purchase, or the components of a student's contact information, or the features of a product in inventory.

Using descriptive and extensible flexfields, you can extend business objects to capture data that wouldn't otherwise be tracked by the application. If you need to add custom fields to a business object to meet your enterprise-specific requirements, configure the flexfield to have one segment for each needed field.

Using key flexfields, you can configure intelligent key codes comprised of meaningful parts according to your business practices. You configure the key flexfield to have one segment for each part that makes up your key code.

Flexfields let you meet enterprise requirements without changing the data model. Different data can be captured on the same database table. Each segment captures a single atomic value, has a name, and maps to a pre-reserved column in the application database.

You can use a flexfield to extend a business object if it has been registered for use on that object. Application developers create a flexfield and register it so that it is available for configuration. Administrators and implementation consultants set up or configure segments and other properties of the available flexfields. End users see flexfield segments as fields or attributes of information displayed in the application user interface. They enter a value for the attribute. The value may be selected from a list of valid values or entered as free-form text that complies with formatting rules.

The following aspects provide an overview of flexfields:

- Accessing flexfields and flexfield management tasks
- Types of flexfields
- Flexfield segments
- Value sets
- Structure and context
- Deployment
- Runtime appearance

Accessing Flexfields and Flexfield Management Tasks

You can view flexfields on a page where they occur using the Highlight Flexfields feature. You can access flexfield management tasks directly from a
highlighted flexfield, through product-specific flexfield management tasks, or by
starting in the Setup and Maintenance Overview page which is available from
the Navigator or the Administration menu.

For lists of flexfields, see assets with the Flexfield: Descriptive, Flexfield:
Extensible, or Flexfield: Key type in Oracle Enterprise Repository for Oracle
Fusion Applications (http://fusionappsoer.oracle.com).

Types of Flexfields
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

For example, in Oracle Fusion Financials, key flexfields represent objects such
as accounting codes and asset categories. Generally, correct operations of a
product depend on key flexfield setup. In Oracle Fusion Payables, a descriptive
flexfield lets you collect custom invoice details fields on an invoices page.
You can implement these fields, which are descriptive flexfield segments, as
custom fields needed beyond the predefined fields.

Segments
Each field that you configure using flexfields is a flexfield segment. Segments
represent attributes of information. They can appear globally wherever the
flexfield is implemented, or based on a structure or context.

You define the appearance and meaning of individual segments when
configuring a flexfield.

A key flexfield segment commonly describes a characteristic of the entity
identified by the flexfield, such as a part number structured to include
information about the type, color, and size of an item. A descriptive flexfield
segment represents an attribute of information that describes a characteristic
of the entity identified on the application page, such as details about a device
containing components, some of which are globally present on the page while
others are contextually dependent on the category of the device.

Value Sets
A value set is a named group of values that can be used to validate the content of
a flexfield segment.

You configure a flexfield segment with a value set that establishes the valid
values that an end user can enter for the segment. You define the values in a
value set, including such characteristics as the length and format of the values.
You can specify formatting rules, or specify values from an application table or
predefined list. Multiple segments within a flexfield, or multiple flexfields, can
share a single value set.
Structure and Context

Key flexfields have structure. Descriptive flexfields and extensible flexfields have context.

Each key flexfield structure is a specific configuration of segments. Adding or removing segments, or rearranging their order, produces a different structure. The database columns on which segments in different structures are based can be reused in as many structures as desired.

Descriptive flexfield segments can be context-sensitive, which means available to an application based on a context value rather than globally available wherever the flexfield appears. A descriptive flexfield context is a set of context-sensitive segments that store information related to the same context value. You define contexts as part of configuring a descriptive flexfield. End users see global segments, as well as any context-sensitive segments that apply to the selected context value.

Extensible flexfield segments are made available to an application based upon a category value. An extensible flexfield context serves as a container for related segments, used to organize the various segments that are applicable to a category value. You define contexts with context-sensitive segments and associate them to categories as part of configuring an extensible flexfield. End users see the segments displayed in subregions, one for each context associated to the selected category value.

In descriptive flexfields and extensible flexfields, the database columns on which context-sensitive segments are based can be reused in as many contexts as desired.

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 appear in the UI based on the flexfield definition at the time of last deployment, rather than the current definition.

Runtime Appearance

In an application user interface, descriptive 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.

Use the Highlight Flexfields command in the Administration menu of the Setup and Maintenance work area to identify the location of the flexfields on the runtime page. Flexfields in highlight mode display an Information icon button to access details about the flexfield, an Edit icon button to manage the flexfield, and an Add Segment icon button to add flexfield segments.

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, in a table, or on their own page or subwindow.

You can use Oracle Composer to edit the layout, position, or other display features of the flexfield segments.
Configuring Flexfields: Overview

Configuring a flexfield ranges from identifying the need for extending a business object with custom attributes to integrating the custom attributes into the deployment. In the case of key flexfields, configuring the flexfield involves identifying value set assignments and determining segment structures.

Overall Process for Configuring Custom Attributes

For descriptive and extensible flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan flexfield validation.
4. Define the attributes by configuring the flexfield segments.
   a. Use the Manage Extensible Flexfields or Manage Descriptive Flexfields tasks, or use the Configure icon button directly on the page where the flexfield is highlighted. For simple configurations, use the Add Segment, Add Context Value, and Edit Segment icon buttons directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.
5. Deploy the flexfield to the mainline to display the custom attributes on the application pages and to make them available for integration with other tools such as Oracle Business Intelligence.
6. Perform the necessary steps to integrate the custom attributes into the technology stack.

A simple configuration is limited to such actions as adding a format-only field or adding a field with a basic list of values.

Overall Process for Configuring Custom Keys

Using key flexfields, you can configure intelligent key codes comprised of meaningful parts according to your business practices. You configure the key flexfield to have one segment for each part that makes up your key code.

For key flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan the flexfield validation.
4. Define the value sets before configuring the key flexfield segments by going to the Manage Value Sets task.
5. Define the key flexfield structures and their segments, and define structure instances for each structure.
   a. Use the Manage Key Flexfields task or the Configure icon button directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.

6. Deploy the flexfield to the mainline to display it on the application pages and to make it available for integration with other tools such as Oracle Business Intelligence.

7. Perform the necessary steps to integrate the flexfield into the technology stack.

**Flexfields at Runtime: Explained**

Many business objects in Oracle Fusion applications have an associated descriptive or extensible flexfield with which you can create custom attributes for the business object. Some business objects have an associated key flexfield for configuring flexible multiple part keys.

The following aspects are important in understanding flexfields at runtime:

- Finding flexfields on a page
- Why no flexfields are on a page

**Finding Flexfields on a Page**

At runtime, the custom attributes you define as extensible and descriptive flexfield segments appear in the application page just like any other attribute. Key flexfields typically appear in the application page as a field with a key flexfield icon, where the field’s value is actually a collection of segments. In some pages, one or more key flexfield segments may be displayed in the page like any other attribute. Thus, when viewing the page in standard mode, in many cases you may not be able to discern which fields are flexfield segments, or whether flexfields are available to configure on the page.

Use the Highlight Flexfields feature to render the page in a special mode that lets you view:

- Where, if any, flexfields are available on your page
- Which, if any, of the fields on your page are flexfield segments rather than out-of-the-box fields

To obtain information about the flexfields on a page, open the page and choose Highlight Flexfields from the Administration menu. Hover over the Information icon button next to the highlighted fields to display information about the flexfield. Choose Unhighlight Flexfields from the Administration menu when you no longer want to see the highlighted flexfields.

When you click the Configure Flexfield icon button for a highlighted flexfield, the applicable Manage Flexfields task is displayed for that flexfield. For simple configurations, you can click the Add Context Value icon button to add a context
value, or click the **Add Segment** or **Edit Segment** icon buttons to add or edit a global segment or a context-sensitive segment that doesn’t require advanced configuration.

**Note**

Not all flexfields are available for creating custom attributes. Consult the product-specific documentation in Oracle Fusion Applications Help to verify whether there are any restrictions on using the flexfield.

---

**Why No Flexfields Are on a Page**

For a flexfield to be available in the page, it must be registered by developers. If a flexfield is available, you may configure segments. The segments appear on the page only after you have successfully deployed the flexfield. For information about registering flexfields, see the Oracle Fusion Applications Developer’s Guide. Some business objects haven’t been designed to support flexfields. For information about how to enable business objects with flexfield capability, see Getting Started with Flexfields in the Oracle Fusion Applications Developer’s Guide.

**Note**

The following Oracle Sales Cloud applications don’t support flexfields:

- Sales
- Marketing
- Customer Center
- Trading Community Architecture
- Order Capture

To add custom attributes to these applications, use Application Composer. For more information, see the "Editing an Object: Explained" section in the Oracle Sales Cloud Extensibility Guide.

---

**Customizing Flexfields Using Page Composer: Explained**

Using Page Composer, you can create customizations to flexfields that are specific to a page.

In Page Composer, to customize:

- Extensible flexfields, open the page in Source view, and look for a region that is bound to an EffContextsPageContainer task flow. This is the container for the extensible flexfield attributes and contexts. To view the flexfield code and identifying information, open the properties panel for the region. To customize any component within the region, select the desired tag and click Edit.

- Descriptive flexfields, open the page in Source view, and look for `<descriptiveFlexfield>` elements. Open the properties panel for the element to view the flexfield code and identifying information. Within
the properties panel, you may customize properties for the global and context-sensitive segments or re-order the segments on the page.

**Accessing Flexfield Management Tasks: Procedures**

You can configure and manage flexfields by highlighting them on an application page and using the available on-screen configuration tools. Alternatively, you can access product-specific flexfield tasks or global flexfield management tasks.

**Accessing Flexfield Management Tasks through the Runtime Page**

You can identify flexfields on the runtime application page where they are implemented.

1. Navigate to an application page.
2. Choose **Highlight Flexfields** from the **Administration** menu in the global area of Oracle Fusion Applications.
3. View the available flexfields highlighted on the page. If any of the fields on the page are custom fields configured as part of a flexfield, they also appear highlighted.
4. To edit a flexfield, use the:
   - **Configure Flexfield** icon button to access the flexfield management task pages for extensive configuration to the flexfield and its segments.
   - **Add Segment** icon button to access the subwindow for adding segments with limited configuration to descriptive flexfields.
   - **Edit Segment** icon button to access the subwindow for limited configuration changes to descriptive flexfield segments.

**Accessing Flexfield Management Tasks through Setup and Maintenance**

Manage flexfields using tasks you access by starting in the Setup and Maintenance Overview page which is available from the Navigator or the Administration menu.

To access tasks for configuring flexfields and value sets, you must be provisioned with roles that entitle you to access the Define Flexfields task list or tasks for managing product-specific flexfields. Contact your security administrator for details. For information about product-specific flexfield tasks, such as Manage Purchasing Descriptive Flexfields, consult the product-specific documentation in Oracle Fusion Applications Help.

To access the flexfield management tasks and search for existing flexfields, perform the following steps:

1. Choose **Setup and Maintenance** from the **Administration** menu in the global area of Oracle Fusion Applications.
2. Search for Define Flexfields in the All Tasks tab.

---

**Tip**
• Use the Business Object parameter to search:
  • Application Key Flexfields, Application Descriptive Flexfields, and Application Extensible Flexfields to find all tasks related to flexfields.
  • Application Flexfield Value Set to find all tasks related to value sets.
• To manage any:
  • Flexfield across all Oracle Fusion Applications products, search for the Define Flexfields task list and access the Manage Descriptive Flexfields, Manage Extensible Flexfields, and Manage Key Flexfields tasks.
  • Value set across all Oracle Fusion Applications products, search for the Define Flexfields task list and access the Manage Value Sets task.

Restriction
If you are configuring key flexfields, search for and access the Manage Value Sets task to set up value sets before accessing the Manage Key Flexfields task.

3. Expand the task list to view the included tasks.
4. Click the Task icon button to open the manage flexfield pages.
5. Search for all or specific flexfields.
6. In the search results, select the flexfield.
7. Use the Edit action to open pages for viewing and configuring the flexfield. Access to managing value sets is available within the tasks for managing descriptive and extensible flexfields.

Note
Access to managing value sets is:
  • Available within the tasks for managing descriptive and extensible flexfields.
  • Not available within the tasks for managing key flexfields. Therefore, configure value sets prior to configuring your key flexfield.

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.

For example, an airline manufacturer might require very specific attributes for their orders that aren’t provided by the out-of-the-box implementation of an
order. Because a flexfield exists for the order business object, you can use it to create and configure the desired attribute.

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 ADF business component (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.

The following aspects are important in understanding how flexfields and Oracle Fusion Applications architecture work together:

- Integration
- Deployment
- Import and Export
- Runtime
• Patching

Integration
The attributes that you add by configuring flexfields are available throughout the Oracle Fusion Middleware technology stack, allowing the flexfields to be used in user interface pages, incorporated into the service-oriented architecture (SOA) infrastructure, and, in the case of descriptive flexfields, integrated with Oracle Business Intelligence. You identify flexfield segments for integration by the segment's Application Programming Interface (API) name.
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 business intelligence (Oracle Business Intelligence, Extended Spread Sheet Database (ESSbase)).
Flexfield configurations 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.

Importing and Exporting
You can export and import flexfields with a deployment status of Deployed or Deployed to Sandbox across instances of Oracle Fusion Applications using the Setup and Maintenance Overview page. Ensure a flexfield is eligible for migration (by verifying that it has successfully deployed) prior to attempting the migration.

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.

Patching
Flexfield configurations are preserved during patching and upgrading.

Flexfields and Value Sets: Highlights

Before you use flexfields to create custom attributes, you should be familiar with the Oracle Fusion application architecture that enables customization, customization layers, and the customization lifecycle.
In addition to the extensive information in the Oracle Fusion Applications Help about configuring flexfields that are already available for configuration, consider the resources below for adding flexfields to business components and alternatives to flexfields where flexfields cannot be enabled.

To assess the flexfields available in a deployment of Oracle Fusion Applications, see assets of type: flexfield in the Oracle Enterprise Repository at http://fusionappsoer.oracle.com.


**Restriction**

Don’t use Oracle JDeveloper to customize flexfields.

**Before Configuring Flexfields**

You can add custom attributes to a business object using a flexfield, if a flexfield has been registered for that object by developers.

- For information about registering flexfields to business objects, refer to the Oracle Fusion Applications Developer's Guide.
  
  See: Getting Started with Flexfields

  - The user interface page for a business object that a developer extends to support a flexfield must be enabled to display the custom attributes defined by the flexfield.

  See: Adding Descriptive Flexfield UI Components to a Page

  See: Employing an Extensible Flexfield on a User Interface Page

- For Sales, Marketing, Customer Center, Trading Community Architecture, and Order Capture applications, use Application Composer to add custom attributes instead of using descriptive and extensible flexfields. For more information, refer to the Oracle Sales Cloud Extensibility Guide.

  See: Application Composer: Introduction

- If you need to create additional actions on data security resources for extensible flexfields, refer to the Oracle Fusion Middleware Oracle Authorization Policy Manager Administrator's Guide (Oracle Fusion Applications Edition).

  See: Managing Oracle Fusion Applications Data Security Policies

- For information about translating flexfield segment values and values in a value set for different locales, refer to the Oracle Fusion Applications Extensibility Guide.

  See: Translating Flexfield and Value Set Configurations

- For information about displaying translated values of a table-validated value set from the value column for the runtime session's locale, refer to the Oracle Fusion Applications Developer's Guide.
Security

• For an understanding of data security when considering the consequences of applying data security to value sets, refer to the Oracle Fusion Applications Security Guide.

See: Data Security

Deploying Flexfields

• To examine the artifacts of a deployed flexfield configuration that you exported using the exportMetadata WLST command, refer to the Oracle Fusion Applications Extensibility Guide.

See: Exporting Customizations

• For information about synchronizing the updated XML schema definition (XSD) files in the metadata services (MDS) repositories for each service-oriented architecture (SOA) application, refer to the Oracle Fusion Applications Extensibility Guide.

See: Customizing SOA Composite Applications

• For information about incorporating a deployed flexfield into the technology stack, such as customizing the pages, integrating with Oracle Business Intelligence, or integrating into Web Services and service-oriented architecture SOA infrastructure, refer to the Oracle Fusion Applications Concepts Guide.

See: Oracle Fusion Middleware Components

• Oracle ADF services used by SOA composites expose the Web Services Description Language (WSDL) schemas where deployed flexfields are stored.

See: Oracle Fusion Middleware Developer’s Guide for Oracle SOA Suite

Oracle Business Intelligence

• For more information about importing and propagating your flexfield changes, refer to the Oracle Fusion Applications Extensibility Guide.

See: Customizing the Oracle BI Repository (RPD)

• For information about importing business intelligence-enabled flexfield changes into the Oracle Business Intelligence repository, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

Flexfield Management

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 let administrators configure and deploy flexfields. If you deploy a flexfield to a sandbox and decide to apply the configuration to the mainline, select the flexfield in the Manage Flexfields tasks of the Define Flexfields activity and deploy the flexfield in the mainline so that it is available to users.

Consider the following aspects of managing flexfields:

- Registering flexfields
- Planning flexfields
- Configuring flexfields
- Enabling a flexfields segment for business intelligence
- Deploying flexfields
- Optionally changing a flexfield segment's appearance in a user interface page
- Identifying flexfields on a runtime page and troubleshooting
Registering Flexfields

Application development registers flexfields so they are available to administrators and implementation consultants for configuration.

As part of registering a flexfield, application development reserves columns of entity tables for use in flexfields so an enterprise can capture segments to meet their business needs. Many flexfields are registered 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

Before you begin planning flexfields, determine what type is appropriate to your needs, and which business objects are available for customizing 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.

Carefully plan flexfields before configuring them. Before configuring new segments for your flexfields, be sure to plan their implementation carefully.

If you have determined that a business object supports flexfields, and those flexfields have been registered, you can begin planning how to configure the flexfield for your needs. Note the code name of the flexfield you intend to configure so you can find it easily in the Define Flexfield activity.

In some cases you can customize how the flexfield appears on the page.

See Oracle Fusion Applications Help for specific products to determine any restrictions on using product-specific flexfields.

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.

You can configure flexfields using the following methods:

- Go to the manage flexfield tasks in the Setup and Maintenance work area.
- Use the Highlight Flexfields command in the Administration menu while viewing a runtime page.
- Use the **Configure Flexfield** icon button to manage a flexfield, such as change a segment’s sequence number, or configure a flexfield segment’s business intelligence label.
- Use the **Add Segment** icon button to add descriptive flexfield segments and context values, or extensible flexfield segments.

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 identifying information for each segment
• Specifying the display properties such as 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
You can create value sets while creating descriptive and extensible flexfield segments. However, define value sets before configuring key flexfield segments that use them, because you assign existing value sets while configuring key flexfield segments.

When creating table-validated, independent, dependent, or subset value sets while creating descriptive and extensible flexfield segments, you can optionally specify to display the description of the selected value to the right of the segment at runtime.

You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order based on the segments’ sequence numbers. You cannot enter a number for one segment that is already in use for a different segment.

Tip
Consider numbering the segments in multiples, such as 4, 5, or 10, to make it easy to insert new attributes.

A flexfield column is assigned to a new segment automatically, but you can change the assignment before saving the segment. If you need to set a specific column assignment for a segment, create that segment first to ensure that the intended column isn’t automatically assigned to a different segment.

Enabling a Flexfield Segment for Business Intelligence

You can enable 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 when enabling key and descriptive flexfield segments for business intelligence.

For extensible flexfield segments, you can’t assign labels and use equalization to prevent duplication.

Deploying Flexfields

Once you have configured a flexfield, you must deploy it to make the latest definition available to runtime users.

In the Define Flexfields tasks, you can deploy a flexfield using either of the following commands:

• The Deploy Flexfield command to deploy a flexfield to mainline. This is for general use in a test or production environment.
• The Deploy to Sandbox command to deploy a flexfield to sandbox. This is to confirm that the flexfield is correctly configured before deploying it to the mainline.

When using the Add Segment and Edit Segment tools for descriptive flexfields in Highlight Flexfields mode, you can use the Save and Deploy command to save your changes and deploy the flexfield to mainline.

Once deployed, the deployment status indicates the state of the currently configured flexfield relative to the last deployed definition.

Optionally Changing a Flexfield Segment Appearance

The flexfield attributes that 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.

After a flexfield has been deployed to a mainline metadata services (MDS) repository so that it appears on application pages, you can customize it on a per-page basis using Page Composer. For example, you can hide a segment, change its prompt or other properties, or reorder the custom global attributes so that they are interspersed with the core attributes in the same parent layout.

You can only customize the appearance of descriptive and extensible flexfield segments in the UI page using Page Composer once the flexfield is deployed to the mainline.

If the Oracle Fusion applications are running in different locales, you can provide different translations for translatable text, such as prompts and descriptions. Enter translations by signing in using the locale that requires the translated text. You do this by selecting Set Preferences from the Personalization menu in the global area and changing the text to the translated text for that locale.

Identifying Flexfields on a Runtime Page and Troubleshooting

The Highlight Flexfields command in the Administration menu of the Setup and Maintenance work area identifies the location of flexfields on the runtime page by displaying an Information icon button for accessing details about each flexfield.

Even if a descriptive or extensible flexfield hasn't yet been deployed and no segments appear on the runtime page in normal view, the flexfield appears in the Highlight Flexfield view for that page. In the case of descriptive flexfields, the segments as of the last deployment appear. Highlight Flexfields accesses the current flexfield metadata definition.

Use the highlighted flexfield's Edit icon button to manage flexfields directly. Alternatively, note a highlighted flexfield's name to search for it in the tasks for managing flexfields.

To examine a flexfield's configuration, export the deployed artifacts using the exportMetadata WLST command.

For more information on creating flexfields and adding them to a UI page, see the Oracle Fusion Applications Developer's Guide.

For more information about customizing flexfield segment appearance with Oracle Composer, see guidance on customizing existing pages in the Oracle Fusion Applications Extensibility Guide.
Flexfield Segment Properties: Explained

Independent of the value set assigned to a segment, segments may have properties that affect how they are displayed and how they behave.

The following aspects are important in understanding:

- Display properties
- Properties related to segment values
- Properties related to search
- Range validation segments
- Rule validation of segment values
- Naming conventions

Display Properties

The following table summarizes display properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Whether the segment can be used.</td>
</tr>
<tr>
<td>Sequence</td>
<td>The order the segment appears in relation to the other configured segments.</td>
</tr>
<tr>
<td>Prompt</td>
<td>The string to be used for the segment’s label in the user interface.</td>
</tr>
<tr>
<td>Display type</td>
<td>The type of field in which to display the segment.</td>
</tr>
<tr>
<td>Checked and unchecked values</td>
<td>If the display type is check box, the actual values to save. For example, Y and N or 0 and 1.</td>
</tr>
<tr>
<td>Display size</td>
<td>The character width of the field.</td>
</tr>
<tr>
<td>Display height</td>
<td>The height of the field as measured in visible number of lines when the display type is a text area.</td>
</tr>
<tr>
<td>Read only</td>
<td>Whether the field should display as read-only, not editable text.</td>
</tr>
<tr>
<td>Description help text</td>
<td>The field-level description help text to display for the field. Use description help text to display a field-level description that expands on or clarifies the prompt provided for the field.</td>
</tr>
<tr>
<td>Instruction help text</td>
<td>Use instruction help text to provide directions on using the field. If instruction help text is specified, it is displayed in an in-field help note window that appears when users give focus to or hover over the field.</td>
</tr>
</tbody>
</table>
Properties Related to Search

Extensible flexfield segments can be marked as selectively required in search using the indexed property. The indexed property requires end users to enter a value before conducting a search on the attribute represented by the indexed segment. A database administrator must create an index on the segment column representing the indexed attribute.

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. Therefore, 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 of the same data type.
- Both segments must be parts of the same structure in a key flexfield or parts 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 that 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.

Rule Validation of Segment Values

Validation rules on descriptive and extensible flexfield segments determine how an attribute is validated. The value entered for an attribute on a business object may need to match a specified format or be restricted to a list of values. Use a value set to specify the validation rules.

Value set validation is required for global segments and context-sensitive segments, and optional for context segments. In the case of context segments, the application may validate an input value instead of the value set validating the input value against the context segment. However, the application input values must match exactly the valid context segment values. If the context segment values are a superset or subset of the input values, you must assign a table-validated value set or independent value set to validate context values.

When you configure a descriptive flexfield segment, you can specify a constant to use for setting the initial value. The initial value can be an available parameter. For every planned segment, list the constant value or parameter, if any, to use for the initial value.
Naming Conventions

Enter a unique code, name, and description for the segment. These properties are for internal use and not displayed to end users. You can’t change the code after the segment is created.

The Application Programming Interface (API) name is a name for the segment that isn’t exposed to end users. The API name is used to identify the segment in various integration points including web services, rules, and business intelligence. Use alphanumeric characters only with a leading character. For example, enter a code consisting of the characters A-Z, a-z, 0-9 with a non-numeric leading character. The use of spaces, underscores, multi-byte characters, and leading numeric characters isn’t permitted. You can’t change the API name after the segment has been created.

Flexfields Segments: How They Are Rendered

Flexfield segments appear on pages as attributes of business objects.

Settings That Affect Flexfield Segment Display

When you configure flexfield segments, the value you enter for the segment’s display type determines how the segment appears on the runtime page.

How Display Type Values Appear

The figure shows how display types appear at runtime.

In the following figure, identify the display type by letter when referring to the table of descriptions for check box, drop-down list, list of values, pop-up list of values, and radio button group.

A. Check Box

B. Drop-down List

C. List of Values

D. Pop-up List of Values

In the following figure, identify the display type by letter when referring to the table of descriptions for radio button group, text area, text box, and date/time.
The table describes each display type. The Example column refers to the figures above.

<table>
<thead>
<tr>
<th>Display Type</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Box</td>
<td>A</td>
<td>The field is displayed as a check box. If the end user selects the checkbox, the checked value is used. Otherwise, the unchecked value is used.</td>
</tr>
<tr>
<td>Drop-down List</td>
<td>B</td>
<td>The field displays a dropdown list of values from which the end user can select a value.</td>
</tr>
<tr>
<td>List of Values</td>
<td>C</td>
<td>The field displays a dropdown list of values from which the end user can select a value. The user can also click Search to find more values.</td>
</tr>
<tr>
<td>Pop-up List of Values</td>
<td>D</td>
<td>The field displays as a text field with a Search icon button. The end users can type a value in the text field or they can click the Search icon button to open a subwindow for searching.</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>E</td>
<td>The field is displayed as a set of radio buttons. The end user can select one button. Selecting a button deselects any previously selected button in the set.</td>
</tr>
</tbody>
</table>
Text Area  |  F  | The field is displayed as a text area in which the end user can type multiple lines of text. The display width and height specify the visible width and number of lines in the text area, respectively.
--- | --- | ---
Text Box  |  G  | The field is displayed as a text field in which the end user can type a single line of text. The display width controls the width of the text box.
--- | --- | ---
Date/Time  |  H  | The field enables the end user to enter a date if the data type is Date, or a date and time if the data type is Date Time. The user can select the date from a calendar. If the data type is Date Time, the field also displays fields for specifying the hour, minutes, seconds, AM or PM, and time zone.
--- | --- | ---
Hidden  |  | The field isn’t displayed.
--- | --- | ---

**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 that 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.

The following aspects are important in understanding how flexfields and value sets work together:

- Defining value sets
- Shared value sets
- Deployment

**Defining Value Sets**

As a key flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfields, you can define value sets when adding or editing a segment.

---

**Caution**

Be sure that changes to a shared value set are compatible with all flexfield segments that use 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 value 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.

**Value Set Values**

To configure custom attributes to be captured on the value set values screen in the Manage Value Sets task, configure the Value Set Values descriptive flexfield. The object’s code is FND_VS_VALUES_B. This flexfield expects the context code to correspond to the value set code. For each value set, you can define a context whose code is the value set code, and whose context-sensitive segments will be shown for the values of that value set. By default the context segment is hidden since it defaults to the value set code and is not expected to be changed.

You can also define global segments that will be shown for all value sets. However, this would be quite unusual since it would mean that you want to capture that attribute for all values for all value sets.

**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 isn’t 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>The 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>----------</td>
<td>-----</td>
<td>----</td>
<td>------------------------</td>
<td>-----</td>
</tr>
<tr>
<td>Constant</td>
<td>Yes</td>
<td>Yes</td>
<td>Default segment value</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>No</td>
<td>The 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>The default segment value is the parameter’s default and derivation value</td>
<td>The 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>The default segment value is the parameter’s default value</td>
<td>The changed parameter default value doesn’t update segment value. Only the changed derivation value updates the segment value.</td>
</tr>
</tbody>
</table>

**Flexfield Usages: Explained**

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.

**Extensible Flexfields**

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.

**Value Sets**

The usage of value sets specifies the flexfields having segments where the value set is assigned.
FAQs for Flexfield Management

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 sign back in again to view flexfield definition changes reflected in the runtime application user interface page.

A flexfield’s deployment status indicates 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.

How can I enable flexfield segments for Oracle Social Network Cloud Service?

Descriptive flexfield segments can be enabled for integration with Oracle Social Network Cloud Service. When you manage Oracle Social Network Objects during setup and maintenance, search for the business object that includes descriptive flexfields, and select the business object attributes that are defined as flexfield segments.

Flexfield Deployment

Flexfield Deployment: Explained

Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. The deployment process adds the custom attributes to the Web Services Description Language (WSDL) schemas that are exposed by Oracle ADF services and that are used by SOA composites. Flexfields are deployed for the first time during the application provisioning process. After you configure or change a flexfield, you must deploy it to make the latest definition available to end users.

If a descriptive flexfield is enabled for business intelligence, the deployment process redeploys the flexfield’s business intelligence artifacts.

You can deploy a flexfield to a sandbox for testing or to the mainline for use in a test or production runtime environment. You can deploy extensible flexfields as a background process.

After deployment, the custom attributes are available for incorporating into the SOA infrastructure, such as business process and business rule integration. For example, you can now write business rules that depend on the custom attributes. You must sign out and sign back in to Oracle Fusion Applications to see the changes you deployed in the runtime.
The following aspects are important in understanding flexfield deployment:

- Deployment Status
- Initial Deployment Status
- Metadata Validations
- Metadata Synchronization
- Deployment as a Background Process

**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 hasn’t been deployed yet. Updates of the metadata definition aren’t applied in the runtime environment yet.</td>
</tr>
<tr>
<td>Patched</td>
<td>The flexfield metadata definition has been modified through a patch or through a data migration action, but the flexfield hasn’t yet been deployed so the updated definition isn’t reflected in the runtime environment.</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 the 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 haven’t 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. The application provisioning process during installation deploys the 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 all flexfield deployment commands. By successfully validating metadata before running the deployment commands, you can avoid failures in the metadata validation phase of a deployment attempt. The deployment process aborts if it encounters an error during the metadata validation phase. Metadata validation results don’t affect the deployment status of a flexfield.

**Metadata Synchronization**

When an extensible or descriptive flexfield is deployed, the deployment process regenerates the XML schema definition (XSD), which makes the custom attributes available to web services and the SOA infrastructure.

After deploying a flexfield configuration, you must synchronize the updated XML schema definition (XSD) files in the MDS repositories for each SOA application.

---

**Note**

To synchronize the updated XSD files in the MDS repositories in Oracle Cloud implementations, log a service request using My Oracle Support at http://support.com/

---

**Deployment as a Background Process**

You can deploy extensible flexfields or incremental changes made to extensible flexfields as a background process. You must use this action to deploy extensible flexfields that have more than 30 categories. You can also use this action if you want to deploy several extensible flexfields, or if you want to continue working in your session without having to wait for a deployment to complete.

**Flexfield Deployment Status: How It Is Calculated**

Flexfield deployment status indicates how the flexfield metadata definition in the Oracle Fusion Applications database relates to the Application Development Framework (ADF) business components generated into a Metadata Services (MDS) repository.

The following aspects are important in understanding how flexfield deployment status is calculated:

- Settings that affect flexfield deployment status
- How deployment status is calculated
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 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 doesn't affect the deployment status. If the flexfield was deployed to a sandbox and hasn't been edited or redeployed 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 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 that the current flexfield definition isn’t deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the Oracle Fusion application likely 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. If the flexfield definition was Deployed before the patch and then a patch was applied, the status changes to Patched. If the flexfield definition was Edited before the patch and then a patch was applied, the status will remain at Edited to reflect that there are still changes (outside of the patch) that aren't yet in effect.

When a deployment attempt fails, 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

Don’t customize flexfield segment display properties using Page Composer 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. When the flexfield is deployed, the ADF business component objects that implement the flexfield in the runtime user interface are generated in the mainline MDS repository from this source.

Deploying a Flexfield to a 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, you 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.

When you deploy a flexfield sandbox, the process generates the name of the flexfield sandbox, and that flexfield sandbox is set as your current active sandbox. When you next sign in to the application, you can see the updated flexfield configurations. The Oracle Fusion Applications global area displays your current session sandbox.

Note

Unlike a standalone sandbox created using the Manage Sandboxes tool, the sandbox deployed for a flexfield contains only the single flexfield. You can manage flexfield sandboxes, such as setting an existing flexfield sandbox as active or deleting it, using the Manage Sandboxes tool.

When you deploy a flexfield to the mainline after having deployed it to the sandbox, the sandbox-enabled flexfield is automatically deleted.

Sandbox MDS Repository Data

The sandbox data lets you test the flexfield in isolation without first deploying it in the mainline where it could be accessed by users.

Warning

Don’t customize flexfield segment display properties using Page Composer 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 and access 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>
<tr>
<td>deleteFlexPatchingLabels</td>
<td>Displays MDS label of flexfield changes for viewing and deleting patching labels.</td>
</tr>
<tr>
<td>validateFlexDeploymentStatus</td>
<td>Displays list containing flexfields that aren’t deployed or failed deployment.</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. For each WLST flexfield command, adding the `reportFormat='xml'` argument returns the report as an XML string.

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
- Using the `deleteFlexPatchingLabels` command
- Using the `validateFlexDeploymentStatus` 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) if it isn’t currently running.

**UNIX:**

```shell
sh $JDEV_HOME/oracle_common/common/bin/wlst.sh
```

**Windows:**

```cmd
wlst.cmd
```

Connect to the server, replacing the user name and password arguments with your WebLogic Server user name and password.

```python
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 doesn't 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', ['enterprise_id'] ['force'])`

In a multi-tenant environment, replace `enterprise_id` with the Enterprise ID to which the flexfield is mapped. Otherwise, replace with 'None' or don’t 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 doesn't 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 isn't 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()

Execute the following command to deploy all flexfields that have either a READY status or an ERROR status.

deployPatchedFlex(mode='RETRY')

Using the deleteFlexPatchingLabels Command

Whenever you deploy flexfield changes to MDS using the deployPatchedFlex() WLST command, an MDS label is created in the format FlexPatchingWatermarkdate+time. Use the deleteFlexPatchingLabels command to inquire about and delete these labels.

From the WLST tool, execute the deleteFlexPatchingLabels () command with no arguments to delete the flexfield patching labels.

To output a list of flexfield patching labels, execute the command with the infoOnly argument, as follows:

deleteFlexPatchingLabels(infoOnly='true')

Using the validateFlexDeploymentStatus Command

The validateFlexDeploymentStatus() WLST command checks the deployment status of all flexfields in an Oracle Fusion Applications deployment.

validateFlexDeploymentStatus()

Use this command to verify that all flexfields in the current instance of provisioned Java EE applications are deployed.

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.

**Manage Value Sets**

**Value Sets: Explained**

A value set is a group of valid values that you assign to a flexfield segment to control the values that are stored for business object attributes.

An end user enters a value for an attribute of a business object while using the application. The flexfield validates the value 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 that changes to a shared value set are compatible with all flexfield segments using the value set.

The following aspects are important in understanding value sets:

- Managing value sets
- Validation
- Security
- Precision and scale
- Usage and deployment

**Managing Value Sets**

To access the Manage Value Sets page, use the Manage Value Sets task, or use the Manage Descriptive Flexfields and Manage Extensible Flexfields tasks for configuring a segment, including its value set. To access the Manage Values page, select the value set from the Manage Value Sets page, and click **Manage Values**. Alternatively, click **Manage Values** from the Edit Value Set page.

**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 doesn’t present a list of valid values to users.

**Note**

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

**Security**

Value set security only works in conjunction with usage within flexfield segments.

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. The value set is the resource secured by data security policies. If a value set is secured, every usage of it in any flexfield is secured. It isn't 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 doesn't determine which descriptive flexfield data is shown upon querying.

Security conditions defined on value sets 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 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**

If the data type of a value set is 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 can assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfield segments, you can create value sets when adding or editing a segment on the runtime page where the flexfield appears.
The following aspects are important in defining value sets:

- Value sets for context segments
- Format-only validation
- Interdependent value sets
- Table validation
- Range
- Security
- Testing and maintenance

**Value Sets for Context Segments**

When assigning a value set to a context segment, you can only use table-validated or independent value sets.

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context's column length. If you use a table value set, the value set cannot reference flexfield segments in the value set’s WHERE clause other than the flexfield segment to which the value set is assigned.

**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, users can enter the value 456 (for a value set with maximum length of three or more), but can’t enter the value ABC. A format only value set doesn’t 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**

Use an independent value set to validate input against a list that isn’t stored in an application table, and not dependent on a subset of another independent value set.

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. 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 don’t need to choose a value for another segment first to have access to the subset value set.

Independent, dependent, and subset value sets require a customized list of valid values. Use the Manage Values page to create and manage a value set’s valid values and the order in which they appear.

Tip

You can customize the Manage Value Sets page to capture additional attributes for each valid value by adding context-sensitive segments in a new context for FND_VS_VALUES_B descriptive field.

Table Validation

Typically, you use a table-validated set when the values you want to use are already maintained in an application table, such as a table of vendor names. Specify the table column that contains the valid value. You can optionally specify the description and ID columns, a WHERE clause to limit the values to use for your set, and an ORDER BY clause.

If you specify an ID column, then the flexfield saves the ID value, instead of the value from the value column, in the associated flexfield segment. If the underlying table supports translations, you can enable the display of translated text by basing the value set’s value column on a translated attribute of the underlying table. You should also define an ID column that is based on an attribute that isn’t language-dependent so that the value’s invariant ID (an ID that doesn’t change) is saved in the transaction table. This allows the runtime to display the corresponding translated text from the value column for the runtime session’s locale.

Table validation lets you enable a segment to depend upon multiple prior segments in the same context structure. You cannot reference other flexfield segments in the table-validated value set’s WHERE clause. That is, the WHERE clause cannot reference SEGMENT.segment_code or VALUESET.value_set_code.

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 doesn’t 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.

**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.

To enable security on a value set, specify a database resource, typically the code value for the value set. Using the Manage Database Security Policies task, specify conditions, such as filters or SQL predicates, and policies that associate roles with conditions. You can use a filter for simple conditions. For more complex conditions, use a SQL predicate.

Value set data security policies and conditions differ from data security conditions and policies for business objects in the following ways:

- You can grant only read access to end users. You cannot specify any other action.

- When defining a condition that is based on a SQL predicate, use VALUE, VALUE_NUMBER, VALUE_DATE, VALUE_TIMESTAMP, or VALUE_ID to reference the value from a dependent, independent, or subset value set. For table value sets, use a table alias to define the table, such as &TABLE_ALIAS category=70.

When you enable security on 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 shouldn't assume anything about the bind variables; it must assume that 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.

**Testing and Maintenance**

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

You cannot manage value sets in a sandbox.

When you change an existing value set, the deployment status for all affected flexfields changes to Edited. You must redeploy all flexfields that use that value set to make the flexfields reflect the changes. In the UI pages for managing value sets, the value set's usages show which flexfields are affected by the value set changes.
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 doesn't require you to provide translated values now, but you cannot change this option if you decide to provide them later.

**Planning Value Sets: Points to Consider**

The value sets you create and configure depend on the valid values on the business object attributes that will use the value set. When creating value sets, you first give the value set a name and description, and then define the valid values of the set.

The following aspects are important in planning value sets:

- List of values
- Plain text input
- Value ranges
- Value format specification
- Security

**List of Values**

You can use one of the following types of lists to specify the valid values for a segment:

- Table column
- Custom list
  - Subset of custom list
- Dependent custom list

If the valid values exist in a table column, use a table value set to specify the list of values. To limit the valid values to a subset of the values in the table, use a SQL WHERE clause. Table value sets also provide some advanced features, such as enabling validation depending on other segments in the same structure.

Use an independent value set to specify a custom set of valid values. For example, you can use an independent value set of Mon, Tue, Wed, and so forth to validate the day of the week. You can also specify a subset of an existing independent value set as the valid values for a segment. For example, if you have an independent value set for the days of the week, then a weekend subset can be composed of entries for Saturday and Sunday.

Use a dependent value set when the available values in the list and the meaning of a given value depend on which independent value was selected for a previously selected segment value. For example, the valid holidays depend on which country you are in. A dependent value set is a collection of value subsets, with one subset for each value in a corresponding independent value set.

For lists of values type value sets, you can additionally limit the valid values that an end user can select or enter by specifying format, minimum value, and
maximum value. For list of values type value sets, you can optionally implement value set data security. If the Oracle Fusion applications are running in different locales, you might need to provide different translations for the values and descriptions.

Plain Text Input

Use a format-only value set when you want to allow end users to enter any value, as long as that value conforms to formatting rules. For example, if you specify a maximum length of 3 and numeric-only, then end users can enter 456, but not 4567 or 45A. You can also specify the minimum and maximum values, whether to right-justify, and whether to zero-fill. With a format-only value set, no other types of validation are applied.

Value Ranges

You can use either a format-only, independent, or dependent value set to specify a range of values. For example, you might create a format-only value set with Number as the format type where the end user can enter only the values between 0 and 100. Or, you might create a format-only value set with Date as the format type where the end user can enter only dates for a specific year, such as a range of 01-JAN-93 to 31-DEC-93. Because the minimum and maximum values enforce these limits, you need not define a value set that contains each of these individual numbers or dates.

Value Format

Flexfield segments commonly require some kind of format specification, regardless of validation type. Before creating a value set, consider how you will specify the required format.

The following table shows options for validation type and value data type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value data type</td>
<td>Character, Number, Date, Date Time.</td>
</tr>
<tr>
<td>Value subtype</td>
<td>Text, Translated text, Numeric digits only, Time (20:08), Time (20:08:08).</td>
</tr>
<tr>
<td></td>
<td>An additional data type specification for the Character data type for the</td>
</tr>
<tr>
<td></td>
<td>Dependent, Independent, and Format validation types.</td>
</tr>
<tr>
<td>Maximum length</td>
<td>Maximum number of characters or digits for Character data type.</td>
</tr>
<tr>
<td>Precision</td>
<td>Maximum number of digits the user can enter.</td>
</tr>
<tr>
<td>Scale</td>
<td>Maximum number of digits that can follow the decimal point.</td>
</tr>
<tr>
<td>Uppercase only</td>
<td>Lowercase characters automatically changed to uppercase.</td>
</tr>
<tr>
<td>Zero fill</td>
<td>Automatic right-justification and zero-filling of entered numbers (affects</td>
</tr>
<tr>
<td></td>
<td>values that include only the digits 0-9).</td>
</tr>
</tbody>
</table>
Caution

You cannot change the text value data type to a translated text value subtype after creating a value set. If there is any chance you may need to translate displayed values into other languages, choose Translated text. Selecting the Translated text subtype doesn't require you to provide translated values.

Value Sets for Context Segments

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context’s column length. If you use a table value set, the value set cannot reference flexfield segments in the value set’s WHERE clause other than the flexfield segment to which the value set is assigned.

Security

When enabling security on a value set, the data security resource name is an existing value set or one that you want to create. The name typically matches the code value for the value set.

Restriction

You cannot edit the data security resource name after you save your changes.

Table-Validated Value Sets and Bind Variables: Points to Consider

After you assign a value set to a flexfield, you can use bind variables in the WHERE clause.

The following bind variables refer to flexfield elements:

- :{SEGMENT.<segment_code>}
- :{CONTEXT.<context_code>;SEGMENT.<segment_code>}
- :{VALUESET.<value_set_code>}
- :{FLEXFIELD.<internal_code>}
- :{PARAMETER.<parameter_code>}

Segment Code

: {SEGMENT.<segment_code>}

This bind variable refers to the ID or value of a segment where <segment_code> identifies the segment. Where referring to the ID, the value set is ID-validated. Where referring to the value, the value set isn’t ID-validated. The data type of the bind value is the same as the data type of the segment’s column.
For both descriptive and extensible flexfields, the segment must be in the same context as the source segment. The source segment contains the WHERE clause. For descriptive flexfields, if the segment is global, then the source segment must be global.

The segment must have a sequence number that is less than the sequence number of the target segment with this bind variable. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment. For example, the values to select from a CITIES table might depend upon the selected country. If SEGMENT1 contains the country value, then the WHERE clause for the CITIES table might be `<country_code> = : {SEGMENT.SEGMENT1}`.

**Context Code**

`:{CONTEXT.<context_code>;SEGMENT.<segment_code}>`

This bind variable, which is valid only for extensible flexfields, refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of a segment that is in a different context than the target segment (the segment with the WHERE clause).

- The `<context_code>` identifies the context and must be in the same category or in an ancestor category. It cannot be a multiple-row context.
- The `<segment_code>` identifies the segment. The data type of the bind value is the same as the data type of the segment’s column.

**Tip**

The target segment should appear in the UI after the source segment to ensure the source segment has a value. If the target segment’s context is a single-row context, the source and target segments must be on separate pages and the target page must follow the source page.

This bind variable is useful when the set of valid values depends on the value of a segment in another context. For example, the values to select from a CERTIFICATION table for a segment in the Compliance and Certification context might depend on the value of the country segment in the Manufacturing context.

**Value Set Code**

`:{VALUESET.<value_set_code>}`

This bind variable refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of the segment that is assigned to the value set that is identified by the `<value_set_code>`. The data type of the bind value is the same as the data type of the segment's column.

The segment must have a sequence number that is less than the sequence number of the segment with this bind variable. If more than one segment is assigned to the value set, the closest prior matching segment will be used to resolve the bind expression. A matching segment must exist in the current flexfield context.
This bind variable is useful when the set of valid values depends on the value in another segment and that segment code can vary, such as when the value set is used for more than one context or flexfield. For example, the values to select from a CITIES table might depend upon the selected country. If the value set for the segment that contains the country value is COUNTRIES, then the WHERE clause for the CITIES table might be `<county_code> = :{VALUESET.COUNTRIES}`.

**Flexfield Internal Code**

`: {FLEXFIELD.<internal_code>}`

This bind variable refers to an internal code of the flexfield in which the value set is used, or to a validation date. The `internal_code` must be one of the following:

- **APPLICATION_ID** - the application ID of the flexfield in which this value set is used. The data type of `APPLICATION_ID` and its resulting bind value is `NUMBER`.
- **DESCRIPTIVE_FLEXFIELD_CODE** - the identifying code of the flexfield in which this value set is used. The data type of `DESCRIPTIVE_FLEXFIELD_CODE` and its resulting bind value is `VARCHAR2`. Note that you use this string for both descriptive and extensible flexfields.
- **CONTEXT_CODE** - the context code of the flexfield context in which this value set is used. The data type of `CONTEXT_CODE` and its resulting bind value is `VARCHAR2`.
- **SEGMENT_CODE** - the identifying code of the flexfield segment in which this value set is used. The data type of `SEGMENT_CODE` and its resulting bind value is `VARCHAR2`.
- **VALIDATION_DATE** - the current database date. The data type of `VALIDATION_DATE` and its resulting bind value is `DATE`.

**Flexfield Parameters**

`: {PARAMETER.<parameter_code>}`

This bind variable refers to the value of a flexfield parameter where `parameter_code` identifies the parameter. The data type of the resulting bind value is the same as the parameter’s data type.

**Note**

You cannot assign a table value set to a context segment if the WHERE clause uses `VALUESET.value_set_code` or `SEGMENT.segment_code` bind variables.

**Table-Validated Value Set: Worked Example**

In an application user interface, you want to display a list of values that allow customers to enter satisfaction scores. The value column name is 1, 2, 3, 4, 5 and the value column description is Extremely Satisfied, Satisfied, and so on. Users can pick the appropriate value or description which stores the corresponding name so the name value can be used in a calculation expression.
In this case, you can use the FND_LOOKUPS table as the basis for a table-validated value set. The lookup meaning corresponds to the Value Column Name and the lookup description corresponds to the Description Column Name. The properties of the value set are as follows:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM clause</td>
<td>FND_LOOKUPS</td>
</tr>
<tr>
<td>WHERE clause</td>
<td><code>lookup_type = 'CN_XX_CUST_SATISFACT_SCORE'</code></td>
</tr>
<tr>
<td>ID column</td>
<td><code>lookup_code</code></td>
</tr>
<tr>
<td>Value column</td>
<td><code>meaning</code></td>
</tr>
<tr>
<td>Description column</td>
<td><code>description</code></td>
</tr>
<tr>
<td>Enable Flag column</td>
<td><code>enabled_flag</code></td>
</tr>
<tr>
<td>Start Date column</td>
<td><code>start_date_active</code></td>
</tr>
<tr>
<td>End Date column</td>
<td><code>end_date_active</code></td>
</tr>
<tr>
<td>Order by</td>
<td><code>display_sequence</code></td>
</tr>
</tbody>
</table>

After completing this task, you should have created your customer satisfaction value set for the Incentive Compensation page of your implementation project.

**Creating a Value Set Based on a Lookup**

1. From the Setup and Maintenance work area, find the Manage Value Sets task and click the **Go to Task** icon button.
2. On the Manage Value Sets page, click the **Create** icon button.
3. On the Create Value Set page, enter the following values:
   a. In the Value Set Code field, enter `CN_XX_CUSTOMER_SATISFACTION_SCORES`
   b. In the Description field, enter Customer satisfaction score.
   c. In the Module field, select **Search....**
   d. In the Search and Select: Module subwindow, enter Incent in the User Module Name field
   e. Select Incentive Compensation.
   f. Click **OK**.
4. On the Create Value Set page, enter the following values:
   a. In the Validation Type field, select Table.
   b. In the Value Data Type field, select Character.
   c. In the Definition section FROM Clause field, enter FND_LOOKUPS.
   d. In the Value Column Name field, enter DESCRIPTION.
   e. In the Description Column Name field, enter MEANING.
   f. In the ID Column Name field, enter LOOKUP_CODE.
   g. In the Enabled Flag Column Name field, enter ‘Y’.
   h. In the Start Date Column Name field, enter START_DATE_ACTIVE.
   i. In the End Date Column Name field, enter END_DATE_ACTIVE.
j. In the WHERE Clause field, enter LOOKUP_TYPE = 'CN_XX_CUST_SATISFACT_SCORE'.

5. Click Save and Close.

6. In the Manage Value Sets page, click Done.

Adding Attributes to the Manage Value Sets Page: Procedure

For independent, dependent, and subset value sets, you can add attributes to a value set. The attributes appear in the Manage Value Sets UI for capturing additional information about each valid value, such as its purpose.

Typically, these attributes are used to capture internal information. To display attributes on an application page, you must programmatically modify the application to access them.

1. Find the FND_VS_VALUES_B flexfield using the Manage Descriptive Flexfields task.
2. Open FND_VS_VALUES_B for editing.
3. Click Manage Contexts.
4. Create a new context and use the value set code for the context code.
5. Add the new attributes as context-sensitive segments.
6. Deploy FND_VS_VALUES_B to the runtime.
7. Sign out and sign back in.
8. Open the Manage Value Sets page to view the new attributes.

FAQs for Manage Value Sets

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 for a descriptive flexfield segment to be a parameter, which means the entity object attribute to which the chosen parameter is mapped provides the initial default value for the segment.

You can set the default value 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.

**How can I add translated values to a value set?**

Enter a translated value for a locale by signing in using that locale. You do this by specifying the locale by selecting Set Preferences from the Personalization menu in the global area. Sign in and enter the translated values for the value set.

If you don’t provide a translation for a given locale, then the value that was first entered is used for that locale.

**Note**

The value set must be defined with the Translated Text subtype.

---

**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 aren’t treated together as a combination.

All Oracle Fusion Applications business entities that you can access are enabled for descriptive flexfields. Descriptive flexfields are optional. You can choose whether or not to configure and expose segments for the descriptive flexfield defined and registered in your database. For lists of descriptive flexfields, see assets with the Flexfield: Descriptive type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

A descriptive flexfield provides a set amount of segments for an entity. You make the segments of a descriptive flexfield available to end users as individual fields in the application user interface.

**Context**

A descriptive flexfield can have only one context segment to provide context sensitivity.
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.

**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 configure the flexfield by determining the following:

- Which attributes to add using the available segments
- The context values
• 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.

Planning Descriptive Flexfields: Points to Consider

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the runtime page where the flexfield appears. Plan how you will deploy the flexfield for test and production users. Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list, Manage Sandboxes, and Highlight Flexfields for adding and editing flexfield segments.

Planning a descriptive flexfield can involve the following tasks:

1. Identify existing parameters.
2. Identify existing context values and whether the context value is derived.
3. Identify custom attributes and plan the descriptive flexfield segments, segment properties, and structure.
5. Plan initial values.
6. Plan attribute mapping to Oracle Business Intelligence objects.

Identify Existing Descriptive Flexfield Parameters

Some descriptive flexfields provide parameters that can be used to specify the initial value of a descriptive flexfield segment. The parameter is external reference data, such as a column value or a session variable. For example, if a flexfield has a user email parameter, you can configure the initial value for a customer email attribute to be derived from that parameter.

Review the list of available parameters in the Derivation Value field in the Create Segment page for a descriptive flexfield. If you decide to use one of the parameters to set an initial value, select that parameter from the Derivation Value drop-down list when you add the descriptive flexfield segment.

Evaluate Whether the Context Value Is Derived

The context value for a descriptive flexfield might have been preconfigured to be derived from an external reference. For example, if the context is Marriage
Status, then the value might be derived from an attribute in the employee business object. When the context value is derived, you might need to take the derived values and their source into consideration in your plan.

To determine whether the context value is derived, access the Edit Descriptive Flexfield task to view the list of configured context values for the flexfield. The Derivation Value field in the Context Segment region displays a list of available parameters.

If context values have been preconfigured, see Oracle Fusion Applications Help for product-specific information about the use of those values.

Plan the Segments, Segment Properties, and Structure

Identify the custom attributes you need for a business object to determine the segments of the descriptive flexfield. Determine the segment properties such as the prompt, display type, or initial value.

The structure of the descriptive flexfield is determined by its global, context, and context-sensitive segments. Plan a global segment that captures an attribute for every instance of the business object. Plan a context for segments that depend on a condition of situation applying to a particular instance of the business object. Plan context-sensitive segments to capture attributes that are relevant in the context.

There is only one context segment available for descriptive flexfields. If you have more than one group of custom attributes where you could use the context segment, you will have to pick one group over the others, based on your company’s needs and priorities, and add the other custom attributes as global segments.

Plan Validation Rules

Define each segment’s validation rules and check if value sets exist for those rules or you must create new ones. If you must create a value set, you can create it either before configuring the flexfield or while creating or editing a segment.

When determining a segment’s validation rules, consider the following questions:

- What is the data type - character, date, date and time, or number?
- Does the segment require any validation beyond data type and maximum length?
- Should a character type value be restricted to digits, or are alphabetic characters allowed?
- Should alphabetic characters automatically be changed to uppercase?
- Should numeric values be zero-filled?
- How many digits can follow the radix separator of a numeric value? In base ten numerical systems the radix separator is decimal point.
- Does the value need to fall within a range?
• Should the value be selected from a list of valid values? If so, consider the following questions:
  • Can you use an existing application table from which to obtain the list of valid values, or do you need to create a custom list?
  • If you are using an existing table, do you need to limit the list of values using a WHERE clause?
  • Does the list of valid values depend on the value in another flexfield segment?
  • Is the list of valid values a subset of another flexfield segment's list of values?

Plan Initial Values

For every segment, list the constant value or SQL statement, if any, to use for the initial value of the custom attribute.

Plan How Segments Map to Oracle Business Intelligence Objects

If a descriptive flexfield has been enabled for Oracle Business Intelligence, you can make it available for use in Oracle Business Intelligence applications. You can use segment labels to map segments to logical objects. Plan to map segments to logical objects before deploying the flexfield as a way to streamline the import into Oracle Business Intelligence.

Use the Manage Segment Labels page to view preconfigured segment labels. If a segment label doesn’t exist for the logical object, then decide on a code, name, and description in preparation for adding that label. Choose a code, name, and description that will help end users select the correct label.

The mapping equalizes similar context-sensitive attributes that are from different contexts but are mapped to a single logical object. For information about objects in the logical model, see the "Working with Logical Tables, Joins, and Columns" chapter in the Oracle Fusion Middleware Metadata Repository Builder’s Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

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.

In general, configuring descriptive flexfields involves:

1. Creating segment labels for business intelligence enabled flexfields.
2. Configuring global segments by providing identity information, the initial default value, and the display properties.
3. Configuring the context segment by specifying the prompt, whether the context segment should be displayed, and whether a value is required.

4. Configuring contexts by specifying a context code, description, and name for each context value, and adding its context-sensitive segments, each of which is configured to include identifying information, the column assignment, the initial default value, and the display properties.

The following aspects are important in understanding descriptive flexfield management:

- Segments
- Adding Segments to a Highlighted Flexfield
- Usages
- Parameters
- Delimiters
- Initial Values
- Business Intelligence

**Segments**

You can assign sequence order numbers to global segments and to context-sensitive segments in each context. 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. The value set must be independent or table-validated. If table-validated, the WHERE clause must not use the VALUESET.value_set_code or SEGMENT.segment_code bind variables. The value set must be of data type Character with the maximum length of values being stored no larger than the context's column length.

If you don't 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.

**Adding Segments to a Highlighted Flexfield**

When you highlight flexfields on a runtime page and use an Add Segment icon button to create a segment, the segment code, name, description, table column,
and sequence number are set automatically. If you use an Add Segment icon button to configure descriptive flexfield segments, you cannot use an existing value set. Value sets are created automatically when you add the segments. You can enter the valid values, their descriptions, and the default value or specify the formatting constraints for the value set, such as minimum and maximum values.

Depending on display type, the value set you create with the Add Segment icon button is either an independent value set or a format-only value set. The table shows which type of value set is created depending on the segment display component you select.

<table>
<thead>
<tr>
<th>Display Component</th>
<th>Value Set Created with Add Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check box</td>
<td>Independent</td>
</tr>
<tr>
<td>Drop-down list</td>
<td>Independent</td>
</tr>
<tr>
<td>List of Values</td>
<td>Independent</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>Independent</td>
</tr>
<tr>
<td>Text Field With Search</td>
<td>Independent</td>
</tr>
<tr>
<td>Text box</td>
<td>Format Only</td>
</tr>
<tr>
<td>Text area</td>
<td>Format Only</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Format Only</td>
</tr>
</tbody>
</table>

Tip

After you add a context value, refresh the page to see the new value.

Usages

Descriptive flexfield usages allow for the same definition to be applied to multiple entities or application tables, such as a USER table and a USER_HISTORY table. Descriptive flexfield tables define the placeholder entity where the flexfield segment values are stored once you have configured the descriptive flexfield. When you configure a flexfield, the configuration applies to all its usages.

Parameters

Some descriptive flexfields provide parameters, which are attributes of the same or related entity objects. Parameters are public arguments to a descriptive flexfield. Parameters provide outside values in descriptive flexfield validation. 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. Parameters can be referenced by the logic that derives the default segment value, and by 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.
Initial Values

The SQL statement defining an initial value must be a valid statement that returns only one row and a value of the correct type.

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.

- SQL statement with bind variables. You can use the following bind variables in the WHERE clause of the SQL statement.

  - :\{\text{SEGMENT.}<segment\_code>\}: Identifies a segment in the same context.
  
  - :\{\text{CONTEXT.}<context\_code>\};\text{SEGMENT.}<segment\_code>\}: Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it cannot be a multiple-row context.

  - :\{\text{VALUESET.}<value\_set\_code>\}: Identifies the closest prior segment in the same context that is assigned to the specified value set.

  - :\{\text{FLEXFIELD.}<internal\_code>\}: Identifies a flexfield.

For more information about using bind variables, see the help for value sets.

Business Intelligence

Selecting a global, context, or context-sensitive segment’s BI Enabled checkbox specifies that the segment is available for use in Oracle Business Intelligence.

When the flexfield is imported into Oracle Business Intelligence, the label you selected from the BI Label dropdown list equalizes the segment with segments in other contexts, and maps the segment to the logical object represented by the label.

Enabling Descriptive Flexfield Segments for Business Intelligence: Points to Consider

A descriptive flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segments. When a global, context, or context-sensitive segment is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled flexfield segments:

- Flattening business components to use BI-enabled segments in Oracle BI

- Equalizing segments to prevent duplication and complexity in the flattened component
• Mapping attributes of flattened business components to logical objects in Oracle BI

• Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For example, a user can generate a report that includes attributes added by the descriptive flexfield. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

Flattening

When you deploy a business intelligence-enabled descriptive flexfield, the deployment process generates an additional set of flattened Application Development Framework (ADF) business components in addition to the usual ADF business components and ADF faces runtime artifacts that are generated during deployment. The flattened business components include attributes for business intelligence-enabled segments only. Flattening means each custom column in each context shows up as an attribute in an Oracle Business Intelligence folder.

Flattened components include one attribute for the BI-enabled context-segment, and one attribute for each business intelligence-enabled global segment. For BI-enabled context-sensitive segments, consider the following:

• If you assigned a label to the segment, the flattened components include an additional single attribute representing segments with that label.

• If you didn’t assign a label, the flattened components include a discrete attribute for each BI-enabled context-sensitive segment in each context.

Mapping to Logical Objects in Business Intelligence

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence.

If you assign a label to any set of context-sensitive segments that serve the same purpose in different contexts, you can consolidate or equalize the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, a United States context might have a Passport segment and a Canada context might have Visa segment. If you assign the NationalID segment label to both the Passport and Visa segments, they are equalized into the same NationalID attribute in the flattened business component.

Non-labeled context-sensitive segments aren’t equalized across context values, so the flattened components include a separate attribute for each context-sensitive segment for each context value.

Note

It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.
Assign a label to a global segment, context segment, or context-sensitive segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

**Note**

Assigning a label to a context-sensitive segment serves to equalize the attribute across contexts, as well as map the equalized attribute to business intelligence.

**Managing Labels**

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across contexts.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn’t exist in business intelligence, you must manually map the segment to the desired logical object when importing into Oracle Business Intelligence.

In addition, context-sensitive segments without labels cannot be equalized across context values. The flattened components include a separate attribute for each non-labeled context-sensitive segment in each context.

**Importing to Oracle Business Intelligence Repository**

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user.

**Note**

To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

**Tip**

When you import a flexfield into the Oracle Business Intelligence repository, you see both `<name>` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>` attribute contains the value. The `<name>_c` attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.
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 aren’t restricted by the number of columns 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 doesn’t 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.

For lists of extensible flexfields, see assets with the Flexfield: Extensible type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

The following aspects are important in understanding key flexfields:

- Usages
- Categories
- Pages
- Security

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.

Usages include security information for applying no security to user access or enforcing view and edit privileges. Some product-specific extensible flexfields have specialized usage fields beyond those for security.

Categories

You can configure multiple extensible flexfield contexts and group the contexts into categories. All extensible flexfields have at least one category. For some extensible flexfields, you can configure a hierarchy of categories. A child category in the hierarchy can inherit contexts from its parent category.

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 let you 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.

Security

When you configure a flexfield, you set the privileges for a context at the usage level by selecting actions for the view and edit privileges of a context usage.

When an end user performs a search, the user interface displays only the attribute values of the contexts for which the user has view privileges. The user is able to perform a search using all attributes for all contexts, regardless of view privileges.

If end users access a context through a web service, an exception is thrown if they perform an action for which they don’t have privileges.

All extensible flexfields have a base data security resource. Some data security resources for extensible flexfields are preconfigured with actions that you can use to specify access privileges. If no action is preconfigured, a security
administrator can create actions and policies to support access control on the extensible flexfield attributes.

Some extensible flexfields have a translatable option; these flexfields also have a translation data security resource.

**Managing Extensible Flexfields: Points to Consider**

Configuring extensible flexfields involves managing the available flexfields registered with your application database.

The following sequence describes how to configure extensible flexfields:

1. Configuring contexts by creating each context segment and the context-sensitive segments for each context segment, and providing the following for each segments:
   a. Identifying information
   b. Column assignment
   c. Initial default value
   d. Display properties
2. Configuring context usages and usage security by selecting actions to which users should have access:
   - View
   - Edit
   - None, if no special privileges should be enforced.
3. Configuring categories and category details.
4. Associating contexts with a category.
5. Creating logical pages for a category.

The following aspects are important in understanding extensible flexfield management:

- Contexts
- Categories
- Initial values
- Indexed segments
- Pages
- Security
- Deployment

**Contexts**

Each context is displayed to end users as a region containing its context-sensitive segments. You can specify instruction help text to display instructions that
explain how to use the region and its attributes to end users. Instruction help text is displayed at the top of the context region. 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 segment for chapter and a segment for number of pages. Multiple chapters can then be associated with each book in the BOOK table.

For contexts that store multiple rows, you can uniquely identify each row by having the values in each row form a unique key.

If flexfield has a category hierarchy, then you can leverage the hierarchy to reuse contexts for similar entities, such as similar items in a product catalog.

Set the context to translatable so that 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.

Set the context security to give an end user view or edit access to a context. The context’s task flow and region appear in the user interface only for users with view access. With edit access, an end user can edit the context’s attribute values. With no action specified for a usage, no special privileges are enforced through the context’s configuration.

**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. Extensible flexfields with more than 30 categories must be deployed as a background process.

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. Consider category hierarchies to be 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 various 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 logical pages, and determine the sequence in which the pages appear. This serves to connect the contexts so they will always be presented together and in a particular order 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.

**Initial Values**

The SQL statement defining an initial value must be a valid statement that returns only one row and a value of the correct type.

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use the following bind variables in the WHERE clause of the SQL statement.
  - `{SEGMENT.<segment_code>`: Identifies a segment in the same context.
  - `{CONTEXT.<context_code>;<SEGMENT.<segment_code>`: Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it cannot be a multiple-row context.
  - `{VALUESET.<value_set_code>`: Identifies the closest prior segment in the same context that is assigned to the specified value set.
  - `{FLEXFIELD.<internal_code>`: Identifies a flexfield.

For more information about using bind variables, see the help for value sets.

**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. Commonly, a database administrator (DBA) adds columns to the database index.

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 criteria. No search is performed if an end user enters an attribute that isn’t indexed as a search criterion.

### Pages

Define logical pages to group contexts together in the user interface. For a given category, you may create one or more logical pages. You may add one or more of the category’s associated contexts to each of the category’s logical pages.

You can specify:

- The sequence of the contexts within each page.
- The sequence in which the logical pages appear.
- Instruction help text to display instructions that explain how to use the page to end users. Instruction help text is displayed at the top of the logical page, preceding all of its context regions.

### Security

An extensible flexfield’s base data security resource typically has a name with an _B suffix. The translation data security resource is a view of a translation table that typically has a name with an _VL suffix.

If a flexfield supports the translatable option and has a translation data security resource, make sure that you create the action for the appropriate data security resource.

- If you create a context-specific action for a nontranslatable context, add it to the base data security resource.
- If you create a context-specific action for a translatable context, add it to the translation data security resource.

### Deployment

You can only deploy extensible flexfields using the Manage Extensible Flexfields task. You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can add one after another extensible flexfield to your deployment queue when you deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command.
When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a background deployment process. A flexfield’s offline status is cleared and it’s deployment status updated when the background deployment process has completed.

**Note**

The *Offline Status* column refreshes when you perform a new search in the Manage Extensible Flexfields task.

---

**Configuring an Item Extended Attributes Flexfield: Example**

The Item Extended Attributes flexfield provides segments for extending the Item business object. In the Manage Extensible Flexfields task, you configure your product business object to include a Technical Specifications logical page in the user interface for the Electronics and Computers category of items.

In this example, your configuration of this flexfield groups custom attributes into the following contexts:

- Materials and Substances
- Compliance and Certification
- Voltage

**Scenario**

The following list shows an example plan for custom computer attributes for the Item Extended Attributes flexfield. In this example, the Electronics Information page is inherited from the parent Electronics and Computers category.

- Page: Electronics Information
  - Context: Compliance and Certification, single row
    - ISO 14001 (International Organization for Standardization for an Environmental Management System)
    - ENERGY STAR (energy efficiency guidelines)
    - ROHS (Restriction of the use of certain hazardous substances in electrical and electronic equipment)
  - Context: Voltage, single row
    - Minimum voltage
    - Maximum voltage
    - Current type
  - Context: Materials and Substances, multiple rows
    - Material
    - Contain recyclate
- Percent unit mass
- Page: Computer Information
- Context: Processor Specifications, single row
- Manufacturer
- CPU type
- Processor interface
- Processor class
- Processor speed
- Cores

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>Which extensible flexfield is available for configuring a hierarchy of categories?</td>
<td>Item Extended Attributes flexfield</td>
</tr>
</tbody>
</table>

**Collecting Technical Specifications**

Your product inventory pages for electronics and computers require a technical specifications page. Your product inventory pages for furniture require a furniture specifications page and an assembly instructions page. Items in both the electronics and computer category, and in the furniture category, share attributes for specifying materials and substances.

The figure shows a Technical Specifications logical page in the user interface for the Electronics and Computers category, with attributes in the context of Recovery and Recycling, Compliance and Certification, Operating Conditions, and Materials and Substances. The Materials and Substances context is configured for multiple rows so your users can select all the materials and substances required to make a single product, displayed as attribute values in a table.
Analysis

You use logical pages to arrange how the contexts appear in the user interface. Use a context to store all the materials and substances required to make a single product. You can configure a context to store multiple rows per entity. The multiple rows are displayed in a table, as for the Materials and Substances context.

The Technical Specifications logical page contains the attributes for the four contexts.

- Recovery and Recycling
- Compliance and Certification
- Operating Conditions
- Materials and Substances

In the figure, the Furniture category is configured to include a Furniture Specifications logical page and an Assembly Instructions logical page. The two categories (Electronics & Computers and Furniture) share the Materials & Substances context.

Configure Security for the Item Flexfield Configuration

The following table shows an example of data security policies for the Item flexfield.

<table>
<thead>
<tr>
<th>Data Security Resource</th>
<th>Policy</th>
<th>Role</th>
<th>Actions</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_EFF_B</td>
<td>A</td>
<td>VOLTAGE_SPEC</td>
<td>edit_nontrans_voltage</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>B</td>
<td>COMPLIANCE_SPEC</td>
<td>edit_trans_compliance</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>C</td>
<td>COMPUTER_SPEC</td>
<td>edit_trans_attrs</td>
<td>ComputerCategoryFilter</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>D</td>
<td>TELEVISION_SPEC</td>
<td>edit_trans_attrs</td>
<td>TVCategoryFilter</td>
</tr>
</tbody>
</table>

The following table shows the privileges for three of the flexfield’s contexts.
In this example, anyone can view the contexts' attributes, but the edit privileges are restricted as follows:

- Voltage: Editable only by voltage specialists.
- Compliance and Certification: Editable only by compliance specialists.
- Materials and Substances: Only computer specialists can edit these attributes for items in the computer category. Only television specialists can edit these attributes for items in the TV category.

In this example, the Materials and Substances context is secured by a generic action with a condition applied to restrict access by category. Voltage and Compliance and Certification are secured by actions specific to each context.

FAQs for Manage Extensible Flexfields

Why did the extensible flexfield context not appear at runtime?

If a deployed extensible flexfield context doesn’t appear in the user interface, verify that the context is associated with one of the category’s pages defined for the extensible flexfield.

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 sees 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 aren’t 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.

For lists of key flexfields, see assets with the Flexfield: Key type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

---

Note

For information about specific key flexfields, see the Oracle Fusion Applications Help for the product where the associated business component is implemented.

---

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. Flexfield metadata is 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 combinations 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.

In some applications, different users need to see different segment structures for
the same flexfield. A key flexfield can have multiple structures if registered to
support more than one structure.

The flexfield can display different fields for different end users based on a data
condition in your application data, such as the value of another field entered by
the end user or the user’s role. For example, the correctly formatted local postal
address for customer service inquiries differs based on locale. A postal address
key flexfield could display different segments and prompts for different end
users based on a location condition in your application data, such as the user’s
role or a value entered by the user.

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.

Each structure may have one or more structure instances. 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.

**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 have ten parts, define ten combinations. A valid combination is simply an existing or new combination that can currently be used because it isn’t out of date or disabled, and doesn’t 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 lets you 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. Maintain individual combinations, such as part numbers in the combinations page.

**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>
</tbody>
</table>
Define Applications Core Configuration

Each usage or reference to the key flexfield | Application development
---|---
Structure instance | Administrators and implementation consultants
Other | Administrators and implementation consultants

If your key flexfield or certain usages or references of the key flexfield don’t 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 doesn’t 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 isn’t the underlying table for the foreign key page.

**Planning Key Flexfields: Points to Consider**

Your first step in planning your key flexfields is to determine which key flexfields your application requires.

Your plan should include:

- The purpose of the key flexfield
- The number and length of its available segment columns
- Whether your key flexfield allows more than one structure
- Whether more than one structure must be defined
- The number, order and length of your segments for each structure

Consider the following aspects in planning flexfields:

- Before you begin
- Access to flexfield-related tasks
- Restrictions
- Validation rules for flexfield segments

**Before You Begin**

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the runtime page where the flexfield appears. Plan how you will deploy the flexfield for test and production users.
Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list and Manage Sandboxes.

If you plan to use value sets, create them before configuring the key flexfield. You cannot create value sets for key flexfields at the time that you add and configure key flexfield segments.

**Access to Flexfield-Related Tasks**

To access tasks for configuring flexfields and value sets, you must be provisioned with roles that entitle you to access the tasks in the Define Flexfields task list or tasks for managing product-specific flexfields. Contact your security administrator for details. For information about product-specific flexfield tasks, such as Manage Fixed Assets Key Flexfields, consult the product-specific documentation in Oracle Fusion Applications Help.

**Restrictions**

If you plan to use value sets, create them before configuring the flexfield.

Plan your key flexfield configuration to scale to your enterprise needs. For example, if you expect to disable old cost centers and enable new ones frequently, plan a larger maximum size for your cost center value set so that you can have more available values. One thousand available values for a 3-character value set provides more room for changes than 100 available values for a 2-character value set.

Note the code name of the flexfield you intend to configure so you can find it easily in the Define Flexfield task list or tasks for managing product-specific key flexfields.

In some cases you can customize how the flexfield appears on the page.

See Oracle Fusion Applications Help for specific products to determine any restrictions on using product-specific key flexfields.

**Reporting**

If you want to report on your data by certain criteria or sub-entities, such as account number or project or region, consider making that sub-entity a distinct segment, rather than combining it with another sub-entity, so that you can categorize and report on smaller discrete units of information.

**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.
Don’t change the number, order, and maximum length of segments once you have acquired flexfield data.

**Structure Delimiters**

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 doesn’t conflict with the flexfield data. For example, if your data frequently contains periods, such as in monetary or numeric values, don’t use a period as your segment separator. Any character you expect to appear frequently in your segment values or descriptions isn’t 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.

**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 combinations 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 the 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 pages 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 lets you manipulate 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 enables the key flexfield to behave more like a descriptive flexfield.

A code combination maintenance page or combinations page presents the combinations table. This enables directly creating and maintaining code combinations. The combinations table contains all key flexfield segment columns and a unique ID column.

A typical application has only one combinations page. An application might not have a combinations page if it doesn't 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 that you can reuse a single key flexfield in multiple combinations of the same segments or a subset of those 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 doesn’t 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

At the structure level, differences among structure instances include whether dynamic combination creation is allowed.

At the structure instance level, differences among segment instances 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 can 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. Commonly, a database administrator (DBA) adds columns to the database index.

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 search criteria. No search is performed if an end user doesn’t enter at least one query required attribute as search criteria.

**Tip**

Index the Structure Instance Number column on your combinations table to improve runtime 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 lets end users enter values at runtime that produce new code combinations for the flexfield. If **Dynamic Combination Creation Allowed** isn’t 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 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.

**Cross Validation Rules: Explained**

You can control the creation of new key flexfield code combinations by defining cross validation rules. A cross-validation rule defines validation across segments and enforces whether a value of a particular segment can be combined with specific values of other segments to form a new combination.

The table compares segment validation to cross-segment validation:

<table>
<thead>
<tr>
<th>Type of validation</th>
<th>Type of control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment validation</td>
<td>Controls the values you can enter for a particular segment</td>
</tr>
<tr>
<td>Cross-segment validation</td>
<td>Controls the combinations of values that administrators and end users can create for key flexfields</td>
</tr>
</tbody>
</table>

**Note**

You can use cross-validation rules for any key flexfield that has cross-validation enabled. See the documentation for your key flexfield to determine if it supports cross validation.

Cross-validation rules prevent the creation of combinations with values that shouldn’t coexist in the same combination. For example, your company requires that all revenue accounts must have a specific department. Therefore, account
combinations that have revenue account values, such as all values between 4000 and 5999, must have a corresponding department value other than 000, which indicates no department is specified. You can define cross validation rules that disallow creation of combinations with incompatible segments, such as 4100-000 or 5000-000.

Alternatively, suppose your accounting key flexfield has an Organization segment with two possible values, 01 and 02. You also have a Natural Account segment with many possible values, but company policy requires that Organization 01 uses the natural account values 001 to 499 and Organization 02 uses the natural account values 500 to 999. You can create cross-validation rules to ensure that users cannot create a general ledger account with combinations of values such as 02-342 or 01-750.

The following aspects are important to understanding cross validation rules:

- Rule Definitions
- Enforcement
- Timing

**Rule Definitions**

Cross validation rules consist of the following information:

<table>
<thead>
<tr>
<th>Rule Feature</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Uniquely identifies cross validation rules in a deployment.</td>
</tr>
<tr>
<td>Description</td>
<td>Helps administrators identify the purpose of the rule.</td>
</tr>
<tr>
<td>Error message</td>
<td>Explains why the attempted combination violates the rule.</td>
</tr>
<tr>
<td>Start Date, End Date</td>
<td>Indicates the period of time when the rule is in effect.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Determines whether the rule is enforced.</td>
</tr>
<tr>
<td>Condition filter</td>
<td>Determines the conditions under which an enabled cross validation rule should be evaluated.</td>
</tr>
<tr>
<td>Validation filter</td>
<td>Determines the validation that the rule enforces when that condition is met.</td>
</tr>
</tbody>
</table>

When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created. If the event specified in the condition filter isn’t applicable, then the combination is considered to pass the rule and the rule won’t be evaluated even if it is enabled.

**Note**

If you don’t specify any statement in the condition filter, then the condition is always true and the rule is always evaluated.

**Enforcement**

Cross-validation prevents creation of invalid combinations by administrators using maintenance pages and end users using dynamic insertion in foreign key pages.
Enabled rules are enforced when there is an attempt to create a new combination of segment values. Disabled rules are ignored. Deleting the rule has the same effect, but you can re-enable a disabled rule.

**Timing**
When users attempt to create a new combination, the key flexfield evaluates any cross-validation rules that are enabled and in effect.

**Warning**
Cross-validation rules have no effect on combinations that already exist. The flexfield treats as valid any existing invalid combinations that pre-date the rule.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, manually disable those combinations using the combinations page for that key flexfield.

When defining a cross-validation rule, specify a start and end date to limit the time when the rule is in effect. The rule is valid for the time including the From and To dates.

**Cross Validation Rules: Points to Consider**

When you need key flexfield combinations of segment values validated across segments, you can optimize your cross validation rules to improve the experience of administrators and end users.

Consider the following when defining cross validation rules:

- Filters
- Rule Complexity
- Maintenance

**Filters**

A cross validation rule includes a condition filter and a validation filter.

The rule is evaluated using the following logic: If the condition filter is satisfied, then validate that the validation filter is satisfied.

1. The condition filter describes the event under which the rule will be evaluated. If the event specified in the condition filter isn’t applicable, then the rule won’t be evaluated even if it is enabled.

2. When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created.

For example, if your organization has determined that a certain company value, Operations, cannot use a specific cost center, Marketing, you can define a cross validation rule to validate your combinations.

1. The rule evaluates the company condition filter.

2. When company is equal to Operations, the rule evaluates the cost center validation filter.
3. When cost center is equal to Marketing, the rule prevents a combination from being created.

4. The error message you defined for the rule displays to inform the user that the attempted combination violates the rule.

**Note**

This rule doesn't affect the creation of combinations with Marketing cost center and company values other than Operations.

**Rule Complexity**

For optimal performance and ease of understanding, define several simple validation rules instead of using one complex rule. Simple validation rules let you provide a more specific error message and are easier to maintain over time.

Avoid rules that control validation across more than two segments, where possible. While you can define cross validation rules that span two or more segments, keep in mind that it becomes more difficult to interpret cross validation error messages and correct invalid key flexfield combinations as your rules encompass more segments.

**Maintenance**

To maintain consistent validation, review existing key flexfields when you update your cross validation rules. Regardless of your current validation rules, Oracle Fusion Applications accept a key flexfield combination if the combination already exists and is enabled. Therefore, to ensure accurate validation, you must review your existing combinations and disable any combinations that don't match the criteria of your new rules.

**Tip**

To keep this type of key flexfield maintenance to a minimum, decide upon your cross validation rules when you first set up your key flexfield structure. Define cross validation rules before creating combinations and before combinations are used in transactions.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross validation rules, disable those combinations using the combinations page.

**Creating a Cross Validation Rule: Example**

Create cross validation rules to prevent specific combinations of segment values in your account combinations, for example, preventing a particular cost center from being combined with a specific company value. Cross validation rules only affect the creation of new account combinations.

**Scenario**

Enter a new cross validation rule to prevent your InFusion America Inc. company value 01 from being combined with your marketing department value...
300 in an account combination. Your company, InFusion America Inc. does not have a marketing department.

1. Navigate to the Manage Cross-Validation Rules task from within your implementation project, and then click the Go to Task icon.

2. Select your InFusion America chart of accounts.

3. Click the Create icon.

4. Specify a unique rule Name, IFAM01, and an optional Description, Do not combine Marketing Department, 300 with InFusion America, company 01.

5. Enter an optional effective From Date of today. Check Enabled.

6. Click the Change filter condition on the Condition Filter. Enter Company equal to 01. The cross validation rule evaluates if Company 01 was entered and if it was entered, then the validation process continues to evaluate the rule.

**Note**

If you do not specify any statement in the condition filter, then the rule is always evaluated.

7. Click on the Change filter condition on the Validation Filter. Enter Cost Center is not equal to 300. When the rule is evaluated, an account combination must contain a cost center other than 300 before it can be created.

8. Enter an Error Message: Cost Center 300 is not allowed with Company 01. The message displays in the relevant user interfaces and processes when an account combination cannot be created because it violates the rule.

9. Click Save and Close.

**Enabling Key Flexfield Segments for Business Intelligence: Points to Consider**

A key flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segment instances. When a segment instance is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled key flexfield segments.

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component
- Mapping attributes of flattened business components to logical objects in Oracle BI
- Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import
the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

**Flattening**

When you deploy a business intelligence-enabled key flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn’t assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

**Mapping to Logical Objects in Business Intelligence**

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence.

If you assign a label to segments that serve the same purpose in different structures, you can consolidate or equalize the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, an organization may have more than one definition of its key accounting flexfield to support different requirements for accounting reporting, or due to chart of accounts definitions from acquired organizations. A US Accounting Flexfield structure may have a segment called Subaccount to track project expenditures. The same type of information may be tracked in a UK accounting flexfield structure with a segment called Project. Equalize these two segments to create a single list of values for reporting.

Non-labeled segments aren’t equalized across context values, so the flattened components include a separate attribute for each segment for each structure.

---

**Note**

It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

---

**Note**

Assigning a label to a segment serves to equalize the attribute across structures, as well as map the equalized attribute to business intelligence.

---

**Managing Labels**

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify
Define Applications Core Configuration

In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across structures.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn't exist in business intelligence, you must manually map the segment to the desired logical object when importing into Oracle Business Intelligence.

In addition, segments without labels cannot be equalized across structures. The flattened components include a separate attribute for each non-labeled segment in each structure.

---

**Note**

Segment labels serve other functions as well, as presented in Key Flexfields: Explained.

---

**Importing to Oracle Business Intelligence Repository**

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user.

---

**Note**

To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

---

**Tip**

When you import a flexfield into the Oracle Business Intelligence repository, you see both `<name>` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>` attribute contains the value. The `<name>_c` attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

---

**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 Combinations Table for Entering Accounts and Employees**

The code combinations table supports entering account information, such as for expense accounts.

The figure shows the origin in the code combinations 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 combinations 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 combinations table.

If dynamic combination creation isn't 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 doesn't 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 is 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.
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 management repository provided by Oracle WebCenter Content Server. Users managing attachments have no real interaction with the repository unless the repository mode is enabled for attachments on specific business objects. In that case, users can share attachments among objects, update attachments by checking them out of and back into 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.

Attachments Troubleshooting: Explained

Attachments UIs for users to add and manage attachments are fully functional as is, and users usually would not encounter issues. If you customize attachments in any way, for example by creating additional attachment categories and implementing data security on them, then some issues might arise.

Issue: Unable to View, Add, Update, or Delete Attachments

Users encounter issues when trying to view attachments or perform actions such as adding attachments.

- Users can no longer see specific attachments that they were previously able to see.
- Likewise, they can no longer update or delete attachments.
- Users get an error stating that they do not have permission to add attachments.

Resolution

Use the Manage Attachment Entities page to ensure that attachment categories are associated to the relevant attachment entity. For example, if users can no longer see attachments for an expense report, then search for the expense report attachment entity and assign all necessary categories to it. You might need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or what categories to assign.

If data security is implemented on the categories for the attachment entity, then verify that the Enable Security check box is selected in the Manage Attachment Entities page for that entity. Make sure that users have a role with the privileges shown in the following table, to view, add, update, or delete attachments with a specific attachment category.

<table>
<thead>
<tr>
<th>Action</th>
<th>Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>Read Application Attachment (FND_READ_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
<tr>
<td>Add or Update</td>
<td>Update Application Attachment (FND_UPDATE_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete Application Attachment (FND_DELETE_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
</tbody>
</table>

For example, if users have the Read Application Attachment privilege for all categories associated with the expense report attachment entity, except the
Receipts attachment category, then they can view all expense report attachments except those created with the Receipts category. Likewise, if users do not have the Update Application Attachment privilege for any attachment categories tied to the expense report attachment entity, then they cannot create any attachments at all for expense reports.

For more information on attachment category data security, see the Oracle Fusion Applications Developer’s Guide.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

**Issue: Missing Attachment Category**

Users can see existing attachments, but the attachments no longer have an attachment category value.

**Resolution**

When the attachment was added, at least one category existed for the corresponding attachment entity, as otherwise the attachment could not have been added. Since then, the entity was edited so that it no longer has any assigned categories, so the user cannot see the category associated with that attachment.

Use the Manage Attachment Entities page to reassign attachment categories to the relevant attachment entity. For example, if users can no longer see the Receipts attachment category for an attachment to an expense report, then search for the expense report attachment entity and assign to it the Receipts category. You might need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or what additional categories to assign.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

**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.
Define Source Systems

Source Systems: Explained

You can set up source systems to enable users to identify the source of the data they are importing into the Oracle Fusion database. 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 types of entities the source system contains, for example, you can specify that a source system will contain trading community member data.

You can configure the following for a source system:

- Source system code, name, and description
- Source system type
- Enable for Items, Trading Community Members, Order Orchestration and Planning, and Assets

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 to create 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, Order Orchestration and Planning, and Assets

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
- Assets

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, Items, and Assets, then the source system can be selected as a data source in the Trading Community Members, Items, and Asset import UIs; however, the source system won’t be able to 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 of the entities in the source system data will be imported. Within the Source System Entities UI, you can choose to allow multiple source references, which allows multiple records from a source system to map to a single trading community record.

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.

Manage Public Unique Identifier Profile Options

Public Unique Identifier: Explained

Oracle Fusion applications generate thousands of rows of data that require unique identifiers (UIDs). These automatically generated, 18-digit numbers are
not easily read or used by humans. A cogent example is an electronic airline ticket whose numbers can be 13 digits long, such as 0162128736572. There are also confirmation or itinerary numbers that relate directly to that ticket that are only six digits long, such as QDLG9S; these can be used interchangeably with the ticket number. These shorter confirmation numbers are much easier to read and process by passengers and employees alike. A separate Public Unique ID (PUID) is also generated by the document sequencing feature and is available for encoding into a more user-friendly, alphanumeric ID.

The Oracle Fusion default setup shows the automatically generated numeric private UIDs and PUIDs for data rows. To use an alphanumeric PUID instead of the delivered numeric, navigate to the Setup and Maintenance task Manage Public Unique Identifier Profile Options to enable the encoding process. There are two profile options to configure the PUID:

- **CRM Public Unique ID String Encoding** controls the characters used in the encoding of the PUID based on a radix, or base number.

- **CRM Public Unique ID Prefix** defines the optional prefix value for the PUID.

Document sequencing typically begins with the number one. You may choose to start the sequences of your PUIDs at a specific value. See the related links below to determine the document sequencing.

### CRM Public Unique ID String Encoding Profile Option

This profile option determines the set of numbers and letters to be used in encoding the PUID generated by the document sequencing feature. The PUID is delivered with the encoding set to none. To enable the encoding of the PUID, choose a radix, or base conversion algorithm. The available values are stored in the lookup type ZCA_PUID_ENCODING. These encoding methods convert the PUID into user readable IDs using alphanumeric characters rather than just numeric digits. For example, if BASE_26 is chosen, only the upper case letters of the alphabet will be used.

There are seven delivered base values. These are not extensible.

<table>
<thead>
<tr>
<th>RADIX</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE_16</td>
<td>Numbers 0-9, letters A-F</td>
</tr>
<tr>
<td>BASE_26</td>
<td>Letters A-Z, Upper Case</td>
</tr>
<tr>
<td>BASE_29</td>
<td>Numbers 2-9, letters BCDFGHJKLMNPQRSTVWXYZ</td>
</tr>
<tr>
<td>BASE_31</td>
<td>Numbers 0-9, letters ABCDEFGHIJKLMNOPQRSTUVWXYZ</td>
</tr>
<tr>
<td>BASE_36</td>
<td>Numbers 0-9, letters A-Z upper case</td>
</tr>
<tr>
<td>BASE_62</td>
<td>Numbers 0-9, letters A-Z, letters a-z</td>
</tr>
<tr>
<td>NONE (default)</td>
<td>No Encoding</td>
</tr>
</tbody>
</table>

### CRM Public Unique ID Prefix Profile Option

A prefix may be defined as a site-level profile option. This profile option is delivered with no prefix defined. When defined, the inserted text is prepended
to the PUID base encoded document sequence value. For example, if the PUID was CLE123 and the Prefix was ORA, the new PUID would be ORACLE123. Any delimiters or separation characters desired must be defined in the prefix itself. Ensure that the prefix length is short enough so that the concatenated PUID and prefix do not exceed the defined field length, usually no more than 30 characters. For example, you may want the records for the pharmaceutical divisions of your company to be denoted with PUIDs and the prefix Pharma- or Pharma1, Pharma2, and so on.

Define Party Usages

Party Usage: Explained

Party usages describe how a party is used in the context of the implementing organization. For example, a person in the business community may be a consumer or contact. Rules can be associated to a party, and these rules determine when a party usage can and cannot be used.

These are the following rules that can be defined for a party usage:

- Assignment rules
- Exclusivity rules
- Incompatibility rules
- Transition rules

Assignment rules

Assignment rules are used to define how the party usage can and cannot be assigned to parties. You can choose to allow unconditional party usage assignment, enabling the assignment to be manually created, or updated, by an administrator.

Alternatively, you can restrict the manual assignment and update of a party usage, which will mean that the party usage can only be created, or updated, by a business event. For example, creating an account for a party record can automatically assign the Customer party usage to that record.

You can also specify whether the party usage assignment can also be created or updated by the assignment of a party relationship. For example, the party relationship Contact Of can assign a party usage of Organization Contact.

Exclusivity rules

Exclusivity rules enable you to restrict party usage assignment, so that between a specified date range the party usage can be the only usage allowed to be assigned to a party. For example, you can set up an exclusivity rule that between January 1 2011 and February 1 2011 parties that have the Manufacturer party usage cannot have any other party usage assigned to them.
Incompatibility rules

You can specify which party usages cannot be assigned concurrently to a party between a defined date range. For example, you can specify that you cannot assign a Sales Account usage to a party with a Sales Prospect usage, during January 1 2011 and December 31 2030.

Transition rules

You are able to define which party usages can transition to the party usage you are creating or editing. Once a party has transitioned to the current party usage, the previous party usage is set with an end date. For example, when a party with the Prospective Partner party usage is assigned the Partner party usage, the Prospective Partner usage is set with an end date.

Party Usage Filter Rules: Explained

Party usage filter rules are a grouping of party usages that are used for searching and filtering in user interfaces. You can specify which party usages you want to include and exclude, and whether the party usage assignments are active or inactive.

For example, an Eligible Customers filter rule can include the Sales Prospect party usage, Sales Account, Legal Entity, and Customer party usages. The Eligible Customers filter rule can then be used in transactional UIs to filter out parties with party usages that are not included in the rule.

Note

Once a party usage filter rule is created you need to compile the filter rule before you can use the filter rule in a User Interface.

FAQs for Define Party Usages

How can I compile a party usage filter rule?

In the Party Usage Filter Rule page click Actions in the header section, and then click Compile Filter Rule.

Define Names

Additional Name Types: Explained

Define additional name types to capture alternative names for parties.
For example, create an additional name type NICK_NAME to capture the nick names of parties of type Person.

While creating an additional name type, specify the party type to which it applies. From then on, the additional name type will be available as a naming option for all parties of that party type within the deploying company. You can use additional name types to capture language-specific names as well.

Define Identifiers

Identifier Types: Explained

Create additional identifier types to provide extensions to 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. These party types can be Person or Organization, or both. You can also specify whether the value of an identifier type must be unique. For example, the passport number listed under each person’s profile must be unique.

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 Trading Community Member Name and Identifier Setup web service.

Define Relationships

Relationships Types: Explained

A relationship type categorizes relationship phrases and roles that determine the nature and purpose of a relationship. A relationship type defines the aspects of a relationship such as the roles of the subject and object, business rules, and phrases that describe the relationship. A relationship, uses the relationship type, to define how a party is related to another party.

The components of a relationship type are:

- Relationship phrase pair and role pair
- Relationship characteristics

Relationship phrase pair and role pair

A relationship type contains a relationship phrase pair and a role pair. Relationship phrase pair and role pair define the mutual roles that the two parties play in a relationship. For example, for relationship type Employment the
phrase pair contains Employee of and Employer of; and, the role pair contains Employee and Employer.

The role pair describes the roles of the entities immaterial of the direction of the relationship. For example, in an Employment relationship Bill has the role of an employee and Oracle Corporation has the role of an employer.

The phrase pair describes the roles of the entities in the relationship. For example, in an Employment relationship if Bill is the subject and Oracle is the object, then the phrase would be Employee of. Similarly, if Oracle is the subject and Bill is the object, then the phrase would be Employer of.

**Relationship characteristics**

Relationship types include characteristics or rules that define how the relationship type can be used.

While defining a relationship type, you can define if the relationship can be:

- Circular
- Related to self

Circular relationships have two parties related in a circle of relationships. For example, you can create a Competitor relationship type to link two organization parties. In this scenario, ABC Corp is a competitor to XYZ Corp, and XYZ Corp is also a competitor to ABC Corp.

You can create a self-relating relationship type, where the subject and the object of the relationship are the same. For example, you can create a Subsidiary relationship type, where a subsidiary organization ABC is linked to a parent organization XYZ. The parent organization XYZ, in turn, could be linked to itself as a subsidiary using the same relationship type.

**Creating Relationship Types: Worked Example**

This example demonstrates how to create a relationship type, and add relationship roles and phrases to the relationship type.

The following table summarizes key decisions while relationship types.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What are the subject and object party types for the relationship?</td>
<td>The subject type and object type is organization.</td>
</tr>
<tr>
<td>Is the relationship circular?</td>
<td>Yes</td>
</tr>
<tr>
<td>Can the subject related to self in this relationship?</td>
<td>No</td>
</tr>
</tbody>
</table>

You must search for the relationship type to make sure it does not already exist, before you create a relationship type.

**Searching for a relationship type**

1. From the Setup and Maintenance work area, search for the Manage Relationship Types task and click **Go to Task**.
2. On the Manage Relationship Types page, in the Search region, complete the fields, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship Type</td>
<td>Equals</td>
<td>Vendor</td>
</tr>
<tr>
<td>Subject Type</td>
<td>Equals</td>
<td>Organization</td>
</tr>
<tr>
<td>Object Type</td>
<td>Equals</td>
<td>Organization</td>
</tr>
</tbody>
</table>

3. Click **Search**.
   Review the search results. Ensure that there is no existing Relationship Type for the criteria.

**Creating a relationship type**
1. In the Search Results region, click the **Create** icon to open the Create Relationship Type page.
2. On the Create Relationship Type page, in the Overview region, enter Vendor in the **Relationship Type** field.
3. Select the **Circular** check box.
4. In the Relationship Roles and Phrases region, click the **Create** icon to open the Create Relationship Role and Phrase Pair dialog box. When you create a relationship with a relationship phrase or role, the reverse direction is automatically created with the other phrase or role in the pair.
5. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Type</td>
<td>Organization</td>
</tr>
<tr>
<td>Subject Phrase</td>
<td>Vendor of</td>
</tr>
<tr>
<td>Subject Role Singular</td>
<td>Vendor</td>
</tr>
<tr>
<td>Subject Role Plural</td>
<td>Vendors</td>
</tr>
<tr>
<td>Object Type</td>
<td>Organization</td>
</tr>
<tr>
<td>Object Phrase</td>
<td>Vendee of</td>
</tr>
<tr>
<td>Object Role Singular</td>
<td>Vendee</td>
</tr>
<tr>
<td>Object Role Plural</td>
<td>Vendees</td>
</tr>
</tbody>
</table>

6. Click **OK**.
7. In the Create Relationship page, click **Save and Close**.

**Editing Relationship Types: Worked Example**

This example demonstrates how to edit a relationship type, including adding relationship roles and phrases to the relationship type, and assigning the relationship to a group.
In this example, you will add a role and phrase pair to the relationship type Employment. You will also add the Employment relationship type to the Party Contacts group.

**Editing a relationship type**

1. From the Setup and Maintenance work area, search for the Manage Relationship Types task and click Go to Task.
2. On the Manage Relationship Types page, in the Search region, enter Employment in the Relationship Type field.
3. Click Search.
4. In the Search Results region, select the Employment relationship type.
5. Click the Edit icon to open the Edit Relationship Type page.
6. On the Edit Relationship Type page, in the Relationship Roles and Phrases region, click the Create icon to open the Create Relationship Role and Phrase Pair dialog box.
7. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Type</td>
<td>Organization</td>
</tr>
<tr>
<td>Subject Phrase</td>
<td>Has contract employees</td>
</tr>
<tr>
<td>Subject Role Singular</td>
<td>Contractor</td>
</tr>
<tr>
<td>Subject Role Plural</td>
<td>Contractors</td>
</tr>
<tr>
<td>Object Type</td>
<td>Person</td>
</tr>
<tr>
<td>Object Phrase</td>
<td>Contract Employee of</td>
</tr>
<tr>
<td>Object Role Singular</td>
<td>Contract Employee</td>
</tr>
<tr>
<td>Object Role Plural</td>
<td>Contract Employees</td>
</tr>
</tbody>
</table>

8. Click OK.
9. In the Relationship Groups region, click the Add Row icon.
10. In the Name drop-down list, select Party Contacts.
11. Click Save and Close.

**Relationship Groups: Explained**

Relationship groups categorize relationship phrase and role pairs for functional purpose. For example, you can assign the relationship phrase Subsidiary Of to a group, and assign the phrase Headquarters Of to a different group.

**Uses of Relationship Groups**

Relationship groups are used to determine which relationship roles and phrases are displayed in specific user interfaces of the Oracle applications. A relationship group can correspond to one or more Oracle Applications user interfaces.
For example, you can configure an Oracle application to display only the relationship types that pertain to the Party Contacts information.

**Creating Relationship Groups: Worked Example**

This example demonstrates how to create a relationship group. Relationship groups are used to categorize relationship roles and phrases, based on the user interfaces they appear in.

In this scenario, you will create group for relationship types that link an organization to an external organization.

**Creating a relationship group**

1. From the Setup and Maintenance work area, search for the Manage Relationship Groups task and click Go to Task.
2. On the Manage Relationship Groups page, in the Search Results region, click the Create icon to open the Create Relationship Group page.
3. Complete the fields, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Related External Organizations</td>
</tr>
<tr>
<td>Code</td>
<td>REL_EXT_ORGS</td>
</tr>
<tr>
<td>Description</td>
<td>Indicates a group of relationships that link the organization to external entities.</td>
</tr>
</tbody>
</table>

4. In the Relationship Roles and Phrases region, click on the Add Row icon to add a relationship role and phrase pair.
5. For each relationship role and phrase pair, complete the fields, as shown in this table.

<table>
<thead>
<tr>
<th>Subject Type</th>
<th>Subject Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Supplier</td>
</tr>
<tr>
<td>Organization</td>
<td>Vendor</td>
</tr>
<tr>
<td>Organization</td>
<td>Competitor</td>
</tr>
</tbody>
</table>

6. Click Save and Close.

**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 Contact Points**

**How Setup Telephone Numbering Plans Fits in with the Other Setup Tasks**

Telephone Numbering Plans allow you to define the telephone number format for a country. You can have multiple formats for a country, depending on the requirements.

**How Telephone Number Plans Setup Fits in with other Setup Tasks**

Phone number format, along with the other country-specific location information such as address and name formats, defines geographic-specific data formats. The Manage Telephone Numbering Plans page allows you to create and manage telephone number plans. Oracle Fusion uses the telephone number formats to display phone number in the appropriate style, and to validate phone numbers.
You can define country-specific components of phone numbers, display format styles, and area codes. Oracle Fusion uses the phone formats to correctly parse phone numbers and determine the correct format style for displaying phone numbers in user interfaces.

Oracle Fusion validates phone numbers for a country, based on the format defined for the country. The phone numbers are also validated against a common set of rules that apply to all countries. You can also enter time zone information for a country, at the country code and area code level. The time zone information entered with phone number is validated against the time information defined for the country.

**Telephone Numbering Plan: Explained**

A telephone numbering plan defines the expected number and pattern of digits for a country phone number. It defines the country-specific components of phone numbers, display format styles, and codes. This includes country codes, international prefixes, area codes within a country, and mobile prefixes. You can specify country-specific mobile prefixes to determine if a phone number is mobile or not.

You can also create multiple telephone number display formats for a country. For example, for the US phone format, you define the phone country code as 01, fixed area code as three numbers in length, and subscriber number as seven digits in length. You can then create a telephone format for domestic display, such as (999) 999-9999. You can also create a telephone format for international display, such as 999-999-9999.

**Updating Telephone Numbering Plans: Explained**

You can update telephone number plans from the Edit Telephone Number Plan page. You cannot edit the country information if the country has area codes, mobile prefixes, or phone formats defined.

In the Edit Telephone Number Plan page, you can:

- Edit Scheme
- Edit Regional Information

**Editing Scheme**

You can edit the country-specific phone information in the Scheme section. You can edit the following fields in the section:

- Trunk Prefix: The code to dial before long distance numbers within the country, for example 1 for the US
- International Prefix: The code to dial before international numbers when calling from the country that you are setting up, for example 011 for the US
- Subscriber Number Length: The length of the subscriber number for countries where the subscriber length is fixed.
• Area Code Length: The length of the area code for countries where the area code length is fixed.

Editing Regional Information

You can edit the regional phone number information for the country in the Regional Information section.

The Regional Information section contains the following tabs:

• Area Codes: Allows you to define area codes that you want to use for the country.

• Mobile Prefixes: Allows you to define mobile prefixes that you want to use for this country. Mobile prefixes contain the area code and possible prefixes of the subscriber number. For example, the mobile prefix of 650-506 consists of the 650 area code and the 506 subscriber number prefix.

• Telephone Number Formats: Allows you to define multiple telephone number display formats for a country.

FAQs for Define Contact Points

Why can't I update the country information section in the Edit Telephone Numbering page?

You cannot update the country information for a country if the regional information such as area codes, mobile prefixes, and phone formats are defined for the country. For example, if area codes, mobile prefixes, and phone formats are defined for the United States, then you cannot edit the country information.

Define Data Formats

Data Formats and Setup Activities: How They Fit Together

Data Formats help you define address formats and name formats within your organization, and assign them to specific geographies. These data formats can be used to capture and validate party name and address information. In order to use geography-based validations for addresses, you must set up geography data in addition to the address formats.

Managing Data Styles

You can define new styles for name and address, which would categorize a name and address style format. A Data Style, such as Mailing Address or Concatenated Name, categorizes name and address style formats. Oracle Fusion uses Data Styles to identify a particular manner or situation for which an entity, such
as name or address, is formatted. The application can be customized so that, for example, only certain name and address styles are displayed in the user interface.

**How Data Formats ensure Data Quality**

Use Data Formats to ensure party address and party name data quality. The format is used to present the address elements in the country specific formatting, and help users during address entry. This also provides the flexibility to specify the scope of address validation. Address formats are linked to the geographic data in order to provide address verification, at the data-import level. For example, when creating or editing an address for a party, selecting a country displays the address format for that country.

The use of Name formats ensures that name components are stored in a decomposed manner. This allows you to construct name representations from components in a variety of different styles. Name formats also allow you to store phonetic names that are required for certain geographies.

Data Formats simplify the storage of party address and party name, by storing them as components. Once you setup the address and name format for a geography, you can use the format to validate address and name data.

**Setup Formats: Explained**

Data Formats determine how names and addresses are displayed in the Oracle Fusion application. Name and address formats vary depending on usage, country, and language. You can create name formats and address formats, customize them to display the address and name components, and assign them to specific locales. This allows you to create data formats for various countries, languages, and usages. Additionally, you can create variations of the formats, with each variation having its own layout. For example, an address format may have variations of a general address, rural address, and military address.

There are two types of data formats.

- **Address format**
- **Name format**

**Address Formats**

Address formats specify the layout of an address, including the address components and their position in the layout. For example, an address format for US postal address can include address, city, state, and zip code. Similarly, the address format for UK would include address, city, county, and post code.

**Name Formats**

Name formats, such as Informal or Mailing, allow you to specify how a name is displayed in the application, based on the usage, language, and country. A name format includes name elements, their position, and formatting. You can create a name style format for either an organization name or a person name.
Creating an Address Style Format: Worked Example

This example shows how to create an address style format for a specified address style.

Specify the address style format name and address style, create the format variation layout, and assign a locale.

Creating the address style format name and specifying the address style

Create the address style format code and name, choose which address style you want to use to define this address style format, and specify if you want this address style format to be the default format for the selected address style.

1. On the Manage Address Formats page, click Actions and then click Create.

2. On the Create Address Style Format page, complete the fields in the Overview section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>CA_POSTAL_ADDR</td>
</tr>
<tr>
<td>Name</td>
<td>Canadian Postal Address Format</td>
</tr>
<tr>
<td>Address Style</td>
<td>Postal Address</td>
</tr>
<tr>
<td>Default</td>
<td>No</td>
</tr>
</tbody>
</table>

3. Click Save.

Creating the format variation layout

Specify the address lines you want in the address and the position of the lines in the address layout. You also want to specify if the address lines will be mandatory, whether the address lines will render in uppercase, and whether there will be blank lines after the address line.

1. In the Format section on the Create Address Style Format page, click on the Format Layout tab.

2. In the Format Variation section, click Actions and then click New.

3. Enter a variation number in the Variation field.

4. In the Format Variation Layout section, click Actions and then click New.

5. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>1</td>
</tr>
<tr>
<td>Position</td>
<td>1</td>
</tr>
<tr>
<td>Prompt</td>
<td>Address line 1</td>
</tr>
<tr>
<td>Address Element</td>
<td>Address line 1</td>
</tr>
</tbody>
</table>
Define Common CRM Configuration

<table>
<thead>
<tr>
<th>Required</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

6. Click **Actions**, then click **New** to create another address line.

7. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>2</td>
</tr>
<tr>
<td>Position</td>
<td>2</td>
</tr>
<tr>
<td>Prompt</td>
<td>City</td>
</tr>
<tr>
<td>Address Element</td>
<td>City</td>
</tr>
<tr>
<td>Required</td>
<td>Yes</td>
</tr>
<tr>
<td>Uppercase</td>
<td>Yes</td>
</tr>
</tbody>
</table>

8. Click **Expand** on the City address line, then enter the value 1 in **Blank Lines Before**.

9. Continue to create as many address lines as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, an address line. In this section you can also create an Attribute Transform Function to change a data value into a different value.

**Assigning a locale**

Specify the relevant countries for this address style format.

1. In the Format section on the Create Address Style Format page, click on the **Locale Assignment** tab.
2. Click **Actions** and then click **New**.
3. Choose a country for the address style format.
4. Continue to add the countries that will use this address style format.
5. Click **Save and Close**.

**Creating a Name Style Format: Worked Example**

This example shows how to create a name style format for a specified name style.

Specify the name of the name style format and name style, create the format variation layout, and assign a locale.

**Creating the name of the name style format and specifying the name style**

You create the name style format code and name, and choose which name style you want to use to define this name style format. You also specify if you want this name style format to be the default format for the selected name style.
1. On the Manage Name Formats page, click **Actions** and then click **Create**.

2. On the Create Name Style Format page, complete the fields in the Overview section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>CONCAT_NAME_US</td>
</tr>
<tr>
<td>Name</td>
<td>United States Concatenated Name Format</td>
</tr>
<tr>
<td>Name Style</td>
<td>Concatenated Name</td>
</tr>
<tr>
<td>Default</td>
<td>No</td>
</tr>
</tbody>
</table>

3. Click **Save**.

**Creating the format variation layout**

Specify the name parts you want in the name, and the position of the name parts in the name layout. You also want to specify if the name parts will be mandatory, define whether the name parts will render in uppercase, and whether there will be blank lines after the name part.

1. In the Format section on the Create Name Style Format page, click on the **Format Layout** tab.

2. In the Format Variation section, click **Actions** and then click **Add Row**.

3. Enter a variation number in the **Variation** field.

4. In the Format Variation Layout section, click **Actions** and then click **Create**.

5. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>1</td>
</tr>
<tr>
<td>Position</td>
<td>1</td>
</tr>
<tr>
<td>Prompt</td>
<td>Person First Name</td>
</tr>
<tr>
<td>Name Element</td>
<td>First Name</td>
</tr>
<tr>
<td>Required</td>
<td>No</td>
</tr>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

6. Click **Actions**, then click **Create** to create another address line.

7. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>2</td>
</tr>
<tr>
<td>Position</td>
<td>2</td>
</tr>
<tr>
<td>Prompt</td>
<td>Person Last Name</td>
</tr>
<tr>
<td>Name Element</td>
<td>Last Name</td>
</tr>
</tbody>
</table>
Define Common CRM Configuration

<table>
<thead>
<tr>
<th>Required</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

8. Continue to create as many name parts as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, a name part. In this section you can also create an Attribute Transform Function to change a data value into a different value.

**Assigning a locale**

Specify the relevant countries for this name style format.

1. In the Format section on the Create Name Style Format page, click on the **Locale Assignment** tab.
2. Click **Actions** and then click **Create**.
3. Choose United States in the **Country** field.
4. Choose American English in the **Language** field.
5. Click **Save and Close**.

**FAQs for Define Data Formats**

**How can I create an address style format layout?**

Click **New** in the Manage Address Formats page and add address lines in the Format Variation Layout section. If you require more than one address style format layout, then you can create a format variation and add a different address format layout.

**How can I create a variation of an Address Style Format?**

You can create different variations of an address style format, with each variation having its own layout. For example, a country’s postal address may have variations such as general postal address, rural address, and military address. When you create an address style format, the layout defined for the format is assigned as the default variation. Click **Add** in the Format Variation region of the Edit Address Style Format page, for the address style format, to add a variation of the Address style. For each address style format variation, you can enter a variation rank to define the priority of the variation.

**How can I create a name style format layout?**

Click **New** in the Manage Name Formats page and add line entries for each part of the full name in the Format Variation Layout section, within the Create Name
Style Format page. If you require more than one name style format layout, then you can create a format variation and add a different name format layout.

**How can I create another variation of a Name Style Format?**

You can create different variations of a name style format, with each variation having its own layout. For example, different regions in a country may have different name style formats. Click **Add** in the Format Variation region in the Edit Name Style Format page, for the name style format, to add a variation of the name style. For each name style format variation, you can enter a variation rank to define the priority of the variation.

**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.
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

Extensibility features are available on the Note object. For more information refer to the article Extending CRM Applications: how it works.
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 happens if I delete a resource?

When you delete a resource, the resource is deleted from the Resource Directory and from the organizations and teams with which the resource was associated. It is always best, though, to check a resource’s usage and remove the resource from all teams and organizations before deletion from the resource directory.

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.

**Primary Resource Organization by Usage Explained**

A resource can belong to multiple organizations, and can be assigned different resource roles for each organization. However, a resource can be assigned to one resource organization that becomes the primary organization for the resource as far as a specific usage is concerned.

A resource organization can be said to be a primary resource organization by usage for a resource if the following criteria are met.

- The resource must be a member of the concerned organization.
- The resource organization must be classified as an organization with the specific usage.

For example, a resource organization can be listed as a primary resource organization for marketing for a particular resource if the resource organization is classified as a marketing organization by usage, and has the concerned resource as a member.

**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.
Organization Memberships and Role Assignments: Explained

When you assign a resource to an organization, you should also specify the role the resource needs to play in the organization. This ensures clarity on the kind of tasks that can be assigned to a resource, apart from defining where the resource is situated within the organization’s hierarchy.

Note

There can be only one manager in an organization. However, a resource can be a member of multiple organizations if needed.

Create an Organization with Sales Usage: Worked Example

This example demonstrates how to create a resource organization with sales usage. You have two choices while creating resource organizations with sales usage:

- Create an organization from an existing department
- Create a new organization

Accessing the Managing Sales and Marketing Organizations Screen

These tasks can be accomplished using the Manage Sales and Marketing Organizations screen in the Setup and Maintenance section. To navigate to this screen, perform the following tasks.

1. Click the Navigator link at the top-left section of the Oracle Fusion screen.
2. Click the Setup and Maintenance link in the Navigator. This displays the Setup and Maintenance screen.
3. If the Manage Sales and Marketing Organization task is listed in the Assigned Implementation tasks, click the Go To Task button against it to access the Manage Sales and Marketing Organizations screen.
4. You can also choose to search for the task in the All Tasks tab.

Creating an Organization from an Existing Department

You need to create an organization with sales usage from an existing department if the concerned department needs to be reused as a sales organization.

1. You can either search for the organization whose usage you wish to modify or click the Create button, select the Create from Existing Department option and specify the name of the organization in the Create Organization screen.
2. Specify the Business Unit to which the organization needs to belong and add Sales Organization under the Organization Usages section of the screen.
Creating a New Organization with Sales Usage

1. Select the **Create New Organization** option in the **Create Organization** screen and click **Next**.
2. Specify the name and business unit of the new organization and select **Sales Organization** as its usage.
3. Click **Finish** to save the new organization with sales usage.

Adding a Sales Organization to a Sales Organization Hierarchy: Worked Example

This example describes how you can add a sales organization to a sales organization hierarchy.

As new sales organizations are created within your company, you need to add these organizations to your company's sales organization hierarchy. To do so, perform the tasks detailed below.

Decide whether to Create a New Version

1. Navigate to the **View Organization Hierarchies** screen in the Resource Manager.
2. Search for and display the sales organization hierarchy you want to edit.
3. Click the **Actions** drop-down button. If the new organization needs to be added to the sales organization hierarchy with immediate effect, select the **Edit this Hierarchy Version** option. If the new organization needs to be displayed in the sales organization hierarchy only after a specific date, select the **Create Hierarchy Version** option. If you choose the latter option, you need to specify the date from which the new sales organization hierarchy version needs to be active.

Add a New Organization to the Sales Organization Hierarchy

1. In the **Edit Organization Hierarchy Version** page, select the node in the sales organization hierarchy where you need to add the new sales organization and click the **Add** button.
2. Search and select the new sales organization in the **Add Tree Node** popup screen. This adds the new organization to the sales organization hierarchy.
3. Click on the new organization in the hierarchy, drag it to the node where it needs to be placed, and release. The organization is now moved to the location you specified within the selected sales organization hierarchy.
4. Click **Save and Close** to save the changes made to the sales organization hierarchy version and exit the screen.

Assigning a Resource to an Organization: Example

Once you have identified a resource, you can assign the resource to organizations within the deploying company. This example explains how you can assign resources to organizations.
Scenario

You need to assign a resource to an organization. It is important to state here that a resource can belong to multiple organizations at the same time. This means that even if chosen resources already belong to organizations, you can assign them to other organizations as needed.

To assign a resource to an organization, perform the following tasks.

2. Display the concerned resource's profile by clicking on the resource's name in the search results.
3. Click the Organizations tab in the Resource Profile Details screen. This displays the list of organizations to which the resource belongs.
4. Click the Create button to display the Create Organization Membership popup screen. Use this screen to search and assign organizations to the concerned resource's profile.
5. Click the Organization field and select the organization you need from the drop-down list displayed. You can also enter part of the name of the organization you need and search for it. You can also search for organizations using a range of search options available in the search screen. Click the OK button to add the organization you need to the Create Organization Membership screen.
6. Click the Add Row button in the Create Organization Membership screen. This adds a row to the screen, enabling you to specify the role the resource needs to play in the organization.
7. Click the Role field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.
8. The From Date field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.
9. The To Date is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.
10. Click the OK button to save your changes. This assigns the resource to the organization you chose. You can add the resource to as many other organizations as needed.

You can also assign resources to organizations by searching for organizations and adding members to the concerned organization. To assign resources to organizations by searching for organizations, perform the following operations.

1. Click the Search field and, from the drop-down list displayed, select Organizations.
2. Search for the organization you need and click on the name of the required organization from the search results. This displays the Organization screen, where all the details of the selected organization are displayed.
3. Click the **Members** tab to edit the list of members assigned to the organization.

4. Click the **Create** button to assign a new resource to the organization. This displays the **Add Resource Member** popup screen.

5. Search for the resource by entering the resource's name in the **Resource** field. From the search results displayed, select the resource you wish to assign to the organization and click **OK**. This displays the **Add Resource Member** popup screen with the resource's name added.

6. Click the **Add Row** button to add a role to the newly-added resource.

7. Click the **Role** field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.

8. The **From Date** field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.

9. The **To Date** is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.

10. Click the **OK** button to save your changes. This assigns the resource to the organization you chose. You can add as many resources to the organization similarly.

**Note**

1. The date from which a resource is a member of an organization must be on or after the date on which the resource was hired.

2. If a resource was previously a member of the selected resource organization with an end date later than the current date, and if the resource was removed from the organization before the end date, you cannot reassign the resource to the same organization.

---

**FAQs for Define Resource Organization Information**

### Can I create a new organization?

Yes. You can create new organizations using the **Manage Sales and Marketing Organizations** option in the Functional Setup Manager or the Oracle Fusion Partner Management user interface. You can also create new resource organizations from the **User Management** interface. You can also use the interface to manage user accounts and roles, and to create employee and partner-member users.

### How can I delete a resource from an organization?

To delete a resource from an organization, you need to delete the concerned resource's membership from the organization.
To manage the membership of an organization, navigate to the **Manage Resource Organization** and select the organization concerned. Once the organization's membership details are listed, you can add, delete or edit them as required.

**What happens if I delete a resource from an organization?**

When you delete a resource from an organization, you effectively delete the concerned resource's membership with the organization. This also removes the resource from the organization hierarchy.

**What's a resource organization hierarchy?**

A resource organization displays the relationships between the resource organizations of the deploying company. The resource organization hierarchy is visible to all resources within the company.

**Can I create a new organization hierarchy?**

No. All organizations within your deploying company are part of a large overarching global hierarchy. Therefore, new organization hierarchies cannot be created. You can, however, move existing organizations or attach new organizations to the existing hierarchy and create a new version of the hierarchy. Once you have created a new version, specify the date from which the new version needs to become active and, on the specified date, your new hierarchy version replaces the older hierarchy.

**Can I copy a resource organization hierarchy?**

No. You can only have one resource organization hierarchy active at a time within a deploying company. If you need to create a hierarchy for an organization, you will need to specify the reporting structure within the concerned organization and then make the organization a part of the existing hierarchy.

**Can I create a new organization hierarchy version?**

Yes. To create a new hierarchy version, navigate to the hierarchy whose new version you wish to create and click on the **Actions** button. From the drop-down list displayed, select **Create Hierarchy Version**. make the changes to the hierarchy as required, specify the dates between which the organization hierarchy needs to be active, and click **Save and Close** to save your changes and exit the screen.
How can I reuse a resource organization in multiple organization hierarchies at the same time?

There is always one resource organization hierarchy for internal resource organizations and also one partner organization hierarchy for each partner company. You can have one active version and multiple inactive versions of resource organization hierarchies. Resource organizations can be included into any of the active or inactive organization hierarchies.

Can I create multiple resources as managers within a resource organization?

No. An organization can have only one manager.

Define Resource Role Information

Define Resource Roles: Explained

Define resource roles to create roles that can be assigned to resources within resource organizations or teams. All resource roles have security privileges associated with them, and when you assign a specific role to resources, they automatically receive access to specific business functions, locations, applications, and data within the company. Resource roles are also required for Oracle Fusion CRM applications, and form the basis of enterprise role provisioning.

The initial data seeded into Oracle Fusion includes resource roles that correspond to the seeded job, or enterprise, roles. Use the Manage Resource Roles task to create additional resource roles or to modify seeded roles. You can access the Manage Resource Roles page by searching for the task in the Navigator > Setup and Maintenance page.

You can 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. You must assign a resource organization to a manager resource, who will have visibility into all opportunities assigned to the members of that resource organization.
- Use the Member flag to make a role a subordinate role in the resource hierarchy.
- Tag roles as Administrator or Lead to indicate the nature of the tasks that resources who are assigned these roles perform within the hierarchy.

Additionally, you can use these flags along with the resource organization hierarchy information to maintain manager-to-manager relationships within
the organization. For example, if Resource A is a manager and heads a sales organization, and Resource B heads his own sales organization, then both Resource A & Resource B will report into Resource C, the Sales Vice President. Similarly, you can create resource organizations and manager resources, and relate managers to higher-level managers, building the resource organization and reporting hierarchies of the company.

**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 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.
Resource Role Assignment: Explained

Resource role assignment is the task of assigning roles to active resources within the deploying company. These roles are previously set up, and have associated security privileges. Thus, when you assign a specific role to resources, they automatically receive access to specific business functions within the company. You can assign different roles to the same resource in different organizations. Thus, the same resource can have access to different business functions depending on the security roles associated with the resource roles assigned to them.

Resource Team Membership and Role Assignment Components: How They Work Together

Resources who are team members can be assigned different roles within the team. These roles do not necessarily need to reflect the roles these resources may play in resource organizations. Depending on the requirements of the tasks assigned to the team, resources may be assigned resource roles, and can accordingly access data related to the tasks in which the team is engaged.

Resource Team Membership
A resource can belong to multiple teams depending on the needs of these teams and the skills that the resource offers. This does not affect the resource’s membership with organizations within the deploying company.

Role Assignment
Resources have specific roles to play in every team to which they are assigned. Each of these roles can be different. Also, these roles can be different from the roles assigned to the same resources in resource organizations. Thus, a resource can be a manager in one team, and a member in another simultaneously.

FAQs for Define Resource Role Information

Can I add more than one resource role to a resource?

Yes. While a resource can have only one resource role within a resource organization, a resource can have multiple roles in different organizations across the deploying company.

Can I assign more than one resource role to a resource organization member?

No. One resource can only have one role within an organization. However, a resource can belong to more than one team, and can have multiple roles in each.
Define Resource Team Information

Resource Setup and Resource Role Setup: How They Fit Together

Setting up resource roles helps you define the various roles within your organization so that they can be assigned to resources.

How Resource Role Setup Simplifies Resource Setup

Use the Resource Role Setup tasks to define specific roles that best reflect the way responsibilities are distributed within the deploying company. If your resource roles are created before you start setting up your resources, you can easily assign roles to resources, simplifying task assignment. Resource roles that are tagged as manager or member roles also determine how eventually the reporting hierarchy gets built.

Later, when you need to assign resources to organizations, resource roles can help you specify the precise roles that resources need to play within the organizations to which you assign them.

Resource roles also simplify security provisioning for resources. You can define security roles corresponding to resource roles using the role mapping interface. Once such mapping rules are setup, you can provision security roles by assigning the corresponding resource roles to resources.

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.

What's a duty role?

A duty role is a set of privileges associated with an aspect of a resource role, and one or more duty roles roll up into a resource role. Basically, resource roles can be
said to comprise a set of duties, and each duty has its own set of privileges. Thus, privileges roll up into duty roles, and duty roles roll up into resource roles.

**Define Products: Define Basic Items**

**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

**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.

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.

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.

Item Imports: 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.

Import Item

The Import Item task 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.

1. Access the Enterprise Storage Server and provide a process name (job definition) such as Item Import Process.

2. Select Setup and Maintenance from the Navigator.

Access the All Tasks tab on the Overview page, and search for the Import Item task with the name of your ESS process definition.

3. 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

4. Click Submit and the Request Number will be displayed.

**Monitoring Import Items**

Select Setup and Maintenance from the Navigator, then access Monitor Item Imports to search for specific Enterprise Storage Server processes and monitor their status in the search results table.

**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.
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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1 dozen = 12 each)</td>
</tr>
<tr>
<td></td>
<td>box</td>
<td>each</td>
<td></td>
</tr>
<tr>
<td></td>
<td>each</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td>pound</td>
<td>gram</td>
<td>454</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1 pound = 454 grams)</td>
</tr>
<tr>
<td></td>
<td>kilogram</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gram</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>hour</td>
<td>second</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1 minute = 60 seconds)</td>
</tr>
<tr>
<td></td>
<td>minute</td>
<td>second</td>
<td></td>
</tr>
<tr>
<td></td>
<td>second</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>cubic feet</td>
<td>cubic inches</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cubic centimeters</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>cubic inches</td>
<td></td>
<td></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 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 systems are two examples of downstream use. 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 transactional item attribute 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 transactional item attribute would not get inherited. Multiple records with same date range can exist if they belong to different domains. For example, the transactional item attribute Memory is associated with a Domain 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 transactional item attribute 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 lifecycle 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.

**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 an item class may be allowed for item creation. For example:

![Diagram of item class hierarchy](image)

This prevents items from being created in Computers and Desktops, but allows 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.

Version control allows new versions to be created for all items in an item class. An integrated workflow definition permits the creation of custom new item request definitions. This workflow definition establishes a multistep process for routing the definition and approval of items. When a new item is created, various people in the organization use the workflow to define aspects of the item, like base operational attributes, user-defined attributes, structures, attachments, categories associated, and organization assignments.

**Note**

Versioning an item and creating a new item request are independent of each other. Versions can be added to a new item request and routed for approval if there is a business need.

When setting up definition steps for a new item request at the item class, you can identify various item details as mandatory, at each step. Definition of entire entity can be made mandatory or just certain attributes. This ensures that the item information required for a downstream step is defined and available for use.
Required attributes can be inherited from parent and assignee access is validated.

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 to 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>Name</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add item button label</td>
<td>The selected value is shown next to the product in the runtime interface.</td>
<td></td>
<td></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>
<td></td>
<td></td>
</tr>
<tr>
<td>Add category enabled flag</td>
<td>Allows buttons to be shown next to the catalog or categories.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Records per page</td>
<td>Number of records to be displayed per page.</td>
<td></td>
<td></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>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>

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>

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.

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 shows 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 relations 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.

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.

When can I use product group mapping?

Use product group mapping in the following situations:
If a product group is reused, find it in the target hierarchy.
Identify the right product group in the target hierarchy when the source product group is not recognized.
Do a reverse mapping by passing the target product group to get the source product group.

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.
Manage Bulk 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 Manage Bulk 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.

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.
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.

Manage Accounting Calendar

Manage Accounting Calendars

The accounting calendar will be used for sales forecast and territory metrics. You need to set up the calendar to match your fiscal calendar and forecasting periods. If you are loading historical data, then you must set the start date to the first date of your historical data.

Important
You cannot change the settings once you start using the application, so consider setting the date to the first date that your company was created. This will allow you to upload historical data later, if necessary.

You can choose the calendar option of your choice: 4-4-5, 4-5-4, 5-4-4, 4 Week, weekly, monthly, quarterly, yearly, or your custom calendar.

Note
• This step is required for territory management and forecasting.
• You can change the values until your first calendar period is open.

Creating the Accounting Calendar

1. In the Setup and Maintenance area, Overview page, All Tasks tab, search for the topic names containing Accounting Calendar

   Manage Accounting Calendars will show up in the results box.

2. 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 because this name will be required in the next topic.

Note
When creating the calendar, the first calendar date must be the first date of the period of the oldest historical data on which you will be reporting. For example, if you select January 1, 2010 as your first calendar date, then you would only be able to enter or import historical data associated with this date and later.
Managing the Calendar Profile Option

Associate the calendar profile with the accounting calendar. The accounting calendar selected here is used as the common calendar.

**Note**

This step is required for Territory Management and Forecasting.

**Caution**

Many features of Oracle Sales use 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 value once it is set.

Managing the CRM Common Calendar Profile Option

The CRM calendar profile option must be associated with the new accounting calendar. Follow these steps:

1. Navigate to the Setup and Maintenance work area, Overview page, All Tasks tab, and search for task names containing Calendar.
2. Locate Manage Calendar Profile Option, and click the Go To Task icon.
3. On the Manage Calendar Profile Option page, locate the ZCA_COMMON_CALENDAR: Profile Values subpage and click the profile option value line.
4. Click the Profile Value list, and select the calendar name that you created when you created the accounting calendar. Click Save and Close.

Define Sales Prediction Configuration

**Oracle Fusion Sales Prediction Engine: Overview**

Oracle Fusion Sales Prediction Engine enables organizations to capture and leverage predictive sales intelligence. Predictive models analyze sales data to evaluate buying patterns. After the evaluation of model results, lead generation can be scheduled to disseminate lead recommendations to users. Each lead recommendation includes win likelihood, average expected revenue, and sales cycle duration.

**Summary of Features**

The key features of Oracle Fusion Sales Prediction Engine include the following:
• Application Home Page: The application home page provides sales analysts with a summary of the prediction model results. Additionally, reports on the dashboard provide overviews of model performance and leads adoption.

• Predictive Model Learning: Model learning uncovers hidden selling patterns in complex business environments. Salespeople can replicate sales success using historical insight generated through model training.

• Rule-based Recommendations: When new products are launched or during initial deployment, historical data is sparse. In such cases, the sales analyst can create customer-, industry-, or product-specific rules to drive the recommendation of new products.

• Higher Lead Adoption Rate: By utilizing a combination of data mining, segmentation, prediction and business rules, Sales Prediction Engine ensures that the recommendations have a higher likelihood of being converted to a win.

• Analyze Recommendation Performance: Built-in analytical reports verify whether the recommendations are being accepted by the sales organization. If adoption is low, then the predictive models can be fine-tuned by selecting different attributes for model learning or editing the rules. Simulation can then be performed to assess the impact of these new changes before publishing new recommendations.

• Usage across CRM Applications: The recommendations generated are integrated with and can be viewed in other CRM applications and features, such as Opportunity Landscape, Customer Center, Territory Management, and Lead Qualification. In Opportunity Landscape, the recommendations can be ranked and qualified as leads after being reviewed. In Customer Center, recommended products display next to a customer, and the rationale for the recommendations is provided in the context of creating a deal for the customer. Territory managers can use the metrics output to set sales targets by territory and assign them to salespeople. The metrics ranking determines whether leads can be qualified in Lead Qualification.

Getting Started with Oracle Fusion Sales Prediction Engine: Prerequisites

This topic helps you get started with Oracle Fusion Sales Prediction Engine. It details the prerequisite tasks and takes you through steps to analyze attributes, run model training, and generate leads.

User Roles

• **Sales Analyst**: Identifies interesting sales trends and customer behavior insights useful to help the overall sales organization to target customers more effectively.

• **Sales Administrator**: Performs ongoing administrative tasks, corrects erroneous or incomplete data, and customizes the application according to business needs. The Sales Administrator is the Oracle Fusion Applications Sales ‘super user’.

Process Flow
Prerequisites

There are important prerequisite tasks that you need to complete. You must check if your data to run model learning is accurate. Running attribute analysis report allows you to view the quality of data and impacts the quality of recommendations generated in Sales Predictor Engine. Once you have results from the attribute analysis report, you can use your judgment to select entities and attributes that are useful for prediction.

You also need to select the products that are recommendable. A company usually has a large portfolio of products but they have the option to make only a subset of them available for model learning, rule creation, and recommendations.

<table>
<thead>
<tr>
<th>Step</th>
<th>Task</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Run attribute analysis report to identify quality of customer attributes.</td>
<td>Sales Analyst</td>
</tr>
<tr>
<td>2</td>
<td>Select model entities and attributes based on the attribute analysis report.</td>
<td>Sales Administrator</td>
</tr>
<tr>
<td>3</td>
<td>Select products suitable for recommendation.</td>
<td>Sales Analyst</td>
</tr>
<tr>
<td>Optional</td>
<td>Set configuration parameters.</td>
<td>Sales Administrator</td>
</tr>
</tbody>
</table>

Running Attribute Analysis Report

You must understand the relative quality of your data to optimize the results of Sales Prediction Engine. Quality data that is well-populated and has good distribution of values ensures best results. If data quality is poor, you can still use it but you must take steps to improve data to maximize results. Use the Attribute Analysis Report to obtain detailed data distribution and importance metrics for each attribute across entities. Based on the analysis report, select the most appropriate attributes for model training. Attributes that have fewer null values and higher importance are good candidates for predictions.

To run attribute analysis report:
1. Log in to Fusion as a sales analyst.

2. In the Navigator menu, select the Recommendations link under the Sales heading.

   The Recommendations page appears.

3. Click the Schedule Attribute Analysis task under the Scheduling heading.

   The Schedule Attribute Analysis page appears.

4. Click the Create icon in the Scheduled and Completed region.

   The Create Attribute Analysis Report page appears.

5. Enter details as required and click the Continue button.

   The Attribute Analysis Report appears with detailed analysis and the importance of each attribute in the selected entities.

### Selecting Entities and Attributes

Sales Prediction Engine generates model training results from historical sales data based on selected model entities and attributes. Based on the findings in the attribute analysis report and your expertise, you can select attributes from each entity that are important predictors for recommendations. You can add or remove attributes easily if the report determines that they add to the predictability of the recommendations. For example, an entity like Customer may have two attributes of high data quality, residential address and annual income. Only if you have a high annual income, you can live in Palo Alto, CA. The analyst may choose to select one of many attributes that have the same impact on prediction to avoid redundancy.

To select entities and attributes:

1. Log in to Fusion Applications as an Administrator.

2. Navigate to the Setup and Maintenance work area by selecting the link in the Navigator menu.

3. On the All Tasks tab, search for the Select Model Entities and Attributes task.

4. Click the Go to Task button for the Select Model Entities and Attributes task.

   The Select Model Entities and Attributes page appears.

5. Select the entities and attributes that you want to include for model training.

   To select attributes, click the respective entity and select attributes from the list of available attributes.

6. Click the Save and Close button.

### Selecting Products for Recommendation

In this step, you select the set of products that will be recommended to customers. Select products based on your organization's business needs. For
example, you may select products based on the sales performance of the past products or expected sales of new products according to the ones you want to promote at a given point. However, after you run model learning and apply rules, from the set of recommendable products, only products that are relevant to a specific customer are recommended.

To select products:

1. In the Navigator menu, select the Recommendations link under the Sales heading.
   The Recommendations page appears.
2. Click the Manage Products for Recommendation task under the Rules and Products heading.
   The Manage Products for Recommendation page appears.
3. Click the Add menu under the Search Results.
   The Browse Catalog page appears.
4. Search for and select the product groups or products for recommendation.
5. Click the Submit button under the Products for Recommendation heading.
   The Manage Products for Recommendation page appears.
6. Click the Done button.

**Setting Configuration Parameters**

The Sales Prediction Engine contains a set of configuration parameters already preset with default values. You can edit these parameters to define how you want the application to function. See the Oracle Fusion Implementation Guide for a list of existing configuration parameters and their function.

The following table lists the important configuration parameters and their function:

<table>
<thead>
<tr>
<th>Configuration Parameter</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayModelBasedRecommendations</td>
<td>True</td>
<td>When this is set to True, model-based recommendations are displayed in consuming applications.</td>
</tr>
<tr>
<td>enableAvgWinRateRecommendations</td>
<td>True</td>
<td>When set to True, recommendations based on average win rates are displayed in consuming applications. This parameter applies only to the Order Capture eligibility rules.</td>
</tr>
<tr>
<td>leadSourceCode</td>
<td>Null</td>
<td>The selected value is updated in the SOURCE_CODE in the lead staging table for leads customization. During implementation, you can customize the ILS with new values.</td>
</tr>
</tbody>
</table>
Number of Top Drivers

| numberOfTopDrivers | 3 | Defines the number of top drivers to return for each recommended product. The values can range from 0-5. |

To set configuration parameters:

1. Log in to Fusion Applications as an Administrator.
2. Navigate to the Setup and Maintenance work area by selecting the link in the Navigator menu.
3. On the All Tasks tab, search for the Manage Recommendations Configuration Parameters task.
4. Click the Go to Task button for the Manage Recommendations Configuration Parameters task.
   The Manage Recommendations Configuration Parameters page appears.
5. Select the configuration parameter that you want to set, and click the Edit icon.
6. Set the desired value and click the Save button.
7. Click the Done button.

Oracle Fusion Sales Prediction Engine: Initial Setup

(On-Premise implementations only.) To run the Oracle Fusion Sales Prediction Engine, you must perform this initial setup task if you have deployed Oracle Business Intelligence Applications and have created the Oracle Business Analytics Warehouse. This topic explains enabling data warehouse for model predictions.

Sales Prediction Engine uses data from the following Data Warehouse entity tables to make accurate predictions:

- Customers
- Assets
- Service Agreements
- Orders

Enabling Data Warehouse

To enable connectivity to Data Warehouse, you must set the value for the profile option.

Note

You can set the value for this profile option using Applications Core Setup, if you have access. The Application Implementation Administrator abstract role provides the necessary access.

1. Sign in to the Oracle Fusion Applications Core Setup.
2. Under the Tasks menu on the left side of the page, click **Manage Administrator Profile Values**. The Manage Administrator Profile Values tab appears.

3. In the **Profile Option Code** field, enter **ZCA_WAREHOUSE_ENABLED_BI** and click **Search**. The profile option appears in the search results.

4. In the Profile Values region, select the result item associated with the searched profile option, and under the Profile Value column, set the value to **Yes**.

5. Click **Save**.

6. Load Assets or Orders tables into the corresponding Data Warehouse table, as applicable.

**Working with Oracle Fusion Sales Prediction Engine**

This topic helps you start working with Oracle Fusion Sales Prediction Engine. It takes you through steps to analyze attributes, run model training, and generate leads.

**User Roles**

- **Sales Analyst**: Identifies interesting sales trends and customer behavior insights useful to help the overall sales organization to target customers more effectively.

**Process Flow**

**Using Sales Prediction Engine**

Once the prerequisite tasks are complete, the sales analyst can use the Sales Prediction Engine to run model training, write rules, and generate leads.

**Summary of Tasks**
Note

All tasks are performed by the sales analyst.

1. Schedule model training.
2. Analyze model training results.
3. Write prediction and eligibility rules.
4. Simulate product recommendation.
5. Generate leads.

Scheduling Model Training

After selecting products, you must schedule model training and check how the model training results look.

1. Click the Schedule Predictive Model Learning task under the Scheduling heading.

   The Schedule Predictive Model Learning page appears.

2. Click the Create icon in the Scheduled and Completed region.

   The Create Predictive Model Training Process page appears.

3. Enter details as required and click the Continue button.

   The model training process is scheduled. View the summary table for the status of the process. Once the process is completed, model training results will be available for review.

   You can now analyze the predictive model to identify products that customers are most likely to buy.

Analyzing Predictive Models

1. Review the Model Insight on the Overview page and details of model results in the Analyze Predictive Models task.

2. You can identify the products with low confidence, likelihood to buy, expected revenue, or high time to close and decide whether you need to eliminate these products from the set of recommendable products or write rules to support them.

   These occurrences may be due to inaccurate or incomplete data and can be reviewed periodically.

Writing Rules

1. Click the Manage Sales Predictor Rules task under the Rules and Products heading.

2. Create prediction or eligibility rules to improve the quality of your recommendations.

   If you want to create a restriction on the products selected using the Manage Products for Recommendation task, you can create a
prediction rule for that product. You can create eligibility rules to further restrict the recommendation of products. If a product is not eligible for purchase by certain customers, you can exclude that product from the recommendations for those customers.

**Note**

Eligibility rules always override model results and prediction rules.

---

**Simulating Product Recommendation**

After you run model training and define rules, you can select a sub-set of customers and conduct a simulation.

1. Click the **Simulate Product Recommendation** task under the **Analysis** heading to simulate recommendations.
2. Review the product recommendations for the selected customers and generate leads if the recommendations look accurate.

**Generating Leads**

1. When you are satisfied with the simulation results, click the **Schedule Sales Leads Generation** task under the **Scheduling** heading to generate leads.

Your leads are generated.

Once you schedule lead generation, the leads will be available in consuming applications such as Opportunity Management and Customer Center.

---

**Model and Rule Entities and Attributes: Explained**

Sales Prediction capitalizes on the power of predictive analytical models to mine and identify patterns in historical data to identify products to sell to your customers.

The decision regarding the selection of entities and attributes is critical. In some cases, the selection of certain entities and attributes may seem logical based on market expertise. For example, customers in certain industries have a stronger affinity for certain products. However in other cases, the model analysis will provide the necessary insight into sales patterns and find attribute values that correlate strongly with sales wins. Additional factors which weigh into the selection of attributes include the availability and accuracy of the attribute data. Finally, attributes that are well-populated may not turn out to be significant in prediction of a sales win.

While selecting attributes, you must not select similar attributes for model training. For example, Annual Revenue and Annual Revenue Category. Your customer’s annual revenue might range from $250,000 to $50 million dollars. However, for efficient management, you decide to target only five customer types based on the Annual Revenue Category such as, Nano ($250,000 dollars to 1 million dollars), Small (1-5 million dollars), and so on. The Annual
Revenue Category uses Annual Revenue for this classification. Therefore, you must use either Annual Revenue or Annual Revenue Category but not both as they are redundant. Similar duplicate attributes could manifest in multiple areas such as Number of Employees and Company Size, Location and Zip Code, and so on.

Sales Prediction also allows the inclusion of expert insight from product management, sales, and marketing operations. You can enforce these expert insights through prediction rules. The same set of entities and attributes are available for both models and rules.

Selecting Model and Rule Entities and Attributes: Examples

Based on model learning insights and your market expertise, you can select entities and attributes that are most important predictors of customers’ likelihood to buy a specific product. To achieve better predictions, you should select only well-populated attributes from the attribute analysis report. Only the selected entities and attributes are available for rules and model learning. The scenarios below provide examples.

Scenario

Account Type Customer

Your company sells a service that appeals to larger customers, 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 that package. You want to know, given a product recommendation, if annual revenue, line of business, customer size, and asset owned are important predictors when it comes to recommending this particular product.

You select the following entities and attributes:

- Account
- Annual Revenue
- Line of Business
- Customer Size Code
- Past Purchased Products or Services
- Assets and Service Contracts

The selected entities and associated attributes can be used to identify buying patterns that have affected wins in the past. Over time, you can further refine the selections based on availability of data and the cost to integrate that data for evaluation.

Scenario

Contact Type Customer

Your company wants to sell a new mobile product that has no historical sales data. You want to create rules to recommend this product to young customers.
You select the following entity and attribute:

- Contact
- Date of Birth

You can now select this attribute among the attributes available in the Create Prediction Rules page to create your rule that targets the new mobile product to customers born before a certain date.

**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 with Cached 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.

**Options for Deploying the Fusion Public Certificate: Explained**

Before using the Oracle Fusion CRM for Microsoft Outlook application, several setup tasks must be performed. One of these tasks is to deploy the Fusion Public Certificate to each user's machine in order to support secure exchange of data between Outlook and Oracle Fusion CRM during synchronization. The Fusion CRM application public certificate is available from the CRM for Microsoft Outlook preference page in the Oracle Fusion Sales application.
User-Managed Deployment using Microsoft Machine Console

If users have sufficient privileges and knowledge to import the certificate themselves, they need to make sure the certificate is imported to both the Personal and Trusted Root Authority certificate stores. This can be done by following these steps:

1. In the Windows Start menu select Run, and type mmc to open the Microsoft Machine Console application.
2. In the console window, select File, then Add/Remove Snap-in.
3. In the Add/Remove Snap-in window, select Add, and then select the Certificates snap-in. Select the Add button to add it.
4. Select the option to manage certificates for My user account and select Finish.
5. Select Close and OK to complete adding the Certificate snap-in.
6. Expand the Certificates - Current User group to review all the certificate stores.
7. Expand the Personal store, right-select the Certificates child folder and select All Tasks - Import.
8. In the next several windows, find the certificate file and complete the import into the Personal certificate store.
9. Repeat step 7 using the Trusted Root Certification Authorities store, and repeat step 8 to import the certificate file.

Automate Deployment with CertMgr.exe and Batch Scripting

Alternatively, to automate the installation of the public certificate into the appropriate certificate stores on the users’ computers, deploy the CertMgr.exe program available from Microsoft in the Windows SDK to users’ computers, along with the certificate file, and a batch script that implements the following commands:

- certmgr.exe -add -c <cert_name> -s root -r localMachine
- certmgr.exe -add -c <cert_name> -r CurrentUser -s My

In the script above, the placeholder for the certificate name would be replaced with the name of the certificate file (for example, oracle.cer).

Customizing the Product Name: Example

You can customize the product name displayed by the Oracle Fusion CRM for Microsoft Outlook client, by modifying the package_res.xml client configuration file.

Use an XML editor to open the package_res.xml client configuration file. Create or modify any of the following attributes, as required:

<str key="app_name">CRM for Microsoft Outlook</str>
<str key="pim_name">Outlook</str>
For example, in the `remote_app_name` attribute, change Oracle Fusion to the name of your company.

### Deploying and Installing Oracle Fusion CRM for Microsoft Outlook: Explained

Oracle Fusion CRM for Microsoft Outlook includes a Microsoft Outlook add-in that must be deployed and installed on each user's computer. The installer file is available from the CRM for Microsoft Outlook preference page in the Oracle Fusion Sales application.

#### User-Managed Deployment

Users can complete the installation by running the Oracle Fusion CRM for Microsoft Outlook.msi file on their computers. The Outlook application must be closed during this process. During the install, the user will specify:

- The install directory
- The Outlook mail profile to use

#### Automated Deployment using Batch Scripting

An alternative to users installing CRM for Microsoft Outlook themselves is for the installer to be deployed to user computers by the administrator using Windows Group policies, Microsoft System Center Configuration Manager (SCCM), or other desktop software deployment mechanisms. In this case, the administrator will access the installer file from the appropriate preference page in the Oracle Fusion Sales application and write a batch script to run the installer with several default parameters, such as the install directory, the mail profile to install to, and all of the elements of the connect string.

The following sample batch script shows how the installer installation can be automated:

```cmd
msiexec /i "Oracle Fusion CRM for Microsoft Outlook.3.00.50.msi"
OL_PROFILE=$DEFAULT FUSION_SERVER_HOST="hostedappserver.com"
FUSION_SERVER_PORT="443" FUSION_SERVER_SUFFIX="outlookEditionConnector/OutlookRequestHandlerService" FUSION_SERVER_PROTOCOL="https" /QR
```

The parameters in the script include the following:

- The name and relative path to the installer file. In the example, the script assumes that the .msi file is in the same directory as the batch script.
- `OL_PROFILE`: This is the name of the user's Outlook mail profile. Besides the mail profile name itself, predefined values can be provided (for example, $DEFAULT and $PREFERRED). When using $DEFAULT, the default mail profile will be selected. When using $PREFERRED, the installer will try to use the default profile first, but if it doesn't satisfy the mail profile configuration requirements (if it doesn't use Cached Exchange Mode or use Personal Folders storage) then the installer will try to use another profile.
- `FUSION_SERVER_HOST`: This is the server name or IP address.
- `FUSION_SERVER_PORT`: This is the port that CRM for Microsoft Outlook is configured to use.
• FUSION_SERVER_SUFFIX: This is the URL suffix for accessing the CRM for Microsoft Outlook Web services. The one provided in the example above is the default deployment path for the CRM for Microsoft Outlook application, and it will typically be used.

• FUSION_SERVER_PROTOCOL: This is either "http" or "https", depending on whether the application is deployed with SSL enabled or not.

Note that the script is using the standard switches provided by the Microsoft Installer executable, msiexec.exe. Documentation of the switches can be reviewed by typing msiexec.exe /? at the command prompt.

Completing the Application Setup after Installation

Once the installer finishes, the first time the user opens Outlook there will be additional dialogs that prompt the selection of various application options. This process is called the First Run Assistant, and each user will specify:

• The Fusion CRM username and password.

• The CRM for Microsoft Outlook connect string. Note that if the installation was completed with the various FUSION_SERVER_* variables, the connect string will be pre-filled and the user will not need to specify those details.

Once the user credentials and connect string are provided, the application will connect to Fusion to download and apply the Outlook configuration available to the user. Once the configuration is applied the user is presented with additional First Run Assistant dialogs to complete the personalization process and perform an initial synchronization. In this second phase of the First Run Assistant, the user will specify:

• Synchronization settings, including the default synchronization frequency and synchronization filters. The application synchronizes data based on synchronization filters, and will automatically initiate a synchronization on the specified frequency.

• Whether to share new Appointments, Contacts, and Tasks in CRM for Microsoft Outlook by default.

• Whether to convert his contact list to Fusion contacts.

Once the user completes the First Run Assistant, the application will begin the first synchronization.

Supported Software for Oracle Fusion CRM for Microsoft Outlook: Explained

Before using the Oracle Fusion CRM for Microsoft Outlook application, several setup tasks must be performed. One of these tasks is to verify each user’s computer has the necessary supported software prior to installing CRM for Microsoft Outlook.

Supported Client Computer Operating Systems

The following are the supported client computer operating systems:

• Microsoft Windows XP SP3 (32 bit) and higher
• Microsoft Vista SP1 (32 and 64 bit) and higher
• Microsoft Windows 7 and higher

**Microsoft Office Versions Supported**
The following are the supported Microsoft Office versions:
• 2007 SP2 and higher
• 2010 and higher

**Supported Outlook Profiles**
The following are the supported Outlook profiles:
• Profile with a Microsoft Exchange e-mail account configured in cached mode with single mailbox. Outlook needs to be run with this profile at least once.
• Profile with an Internet e-mail account. Outlook needs to be run with this profile at least once.

**Note**
CRM for Outlook can only be installed into one of the Outlook profiles.

Depending on what data file is set as the default mail delivery location in the Outlook profile selected for CRM for Outlook installation, it can be either:
• .ost file
• .pst file

**IMAP Not Supported**
Only POP protocol is supported. IMAP is not supported for the following reasons:
• The IMAP protocol has a unique structure. Whether or not a custom folder can be created in this protocol depends on the mail server. The CRM for Outlook platform requires the ability to add custom folders, and needs to be able to synchronize custom CRM objects (for example, accounts or opportunities) with Outlook.
• The IMAP protocol cannot be made the default mail delivery location in Outlook, and the CRM for Outlook platform installs to the default mail delivery location in the profile. Either Exchange mailbox or POP .pest can be made the default mail delivery location. A default mail delivery location is required for the platform, as all Outlook activity processing happens there (for example, meeting participants receive notification from the organizer only in this location).

**Note**
Changing the default mail delivery location in the Outlook profile where CRM for Outlook is installed is not supported because doing so will cause the CRM for Outlook add-in to fail.

**Microsoft Exchange Server Versions Supported**
The following are the supported versions of Microsoft Exchange Server:
• Microsoft Exchange Server 2003
• Microsoft Exchange Server 2007

Other Required Components

Microsoft .NET Framework 3.5 SP1 or higher is required to support the client application.

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.

**Customizing the Support Team E-Mail Address: Example**

When a user clicks **Send Feedback** on the Feedback tab within Oracle Fusion CRM for Microsoft Outlook, a new e-mail message is created, and the e-mail is automatically addressed to the support team. You can customize the support team’s e-mail address by modifying the package_res.xml client configuration file.

Use an XML editor to open the package_res.xml client configuration file. In the following code, change `email_address` to the required e-mail address.

<!-- Feedback page -->
<br>
<str key="support_email">email_address</str>

For example:

<str key="support_email">support@your_company.com</str>

**Making a Field or Form Read-Only: Example**

You can make Oracle Fusion CRM for Microsoft Outlook fields and forms read-only by modifying the forms.js client configuration file.

**Locate the Relevant Section of the Client Configuration File**

Use a JavaScript editor to open the forms.js client configuration file. Find the following code in the configuration file:

```javascript
// LEAD FORM SCRIPTS //
function lead_form(ctx) {
}
```

The example above shows how to make the **Lead** field read-only.

**Making a Field Read-Only**

To make a field read-only, add the following code below the section of code you have just located:

```javascript
ctx.form[control_id].enabled = false;
```
Making a Form Read-Only

To make a form read-only, add the following code below the section of the client configuration file you have just located:

ctx.form.enabled = false;

Customizing Text That Appears in the Oracle Fusion CRM for Microsoft Outlook Client: Example

You can customize the text that appears in the Oracle Fusion CRM for Microsoft Outlook application, by modifying the text strings in the package_res.xml client configuration file.

Use an XML editor to open the package_res.xml client configuration file, and find the text string you want to change. For example, in the following code you want to change the text string "Customer name is required":

<str key="msg_customer_required">Customer name is required.</str>

You change the text string to "Enter a customer name for this organization", and so the code now appears as follows:

<str key="msg_customer_required">Enter a customer name for this organization.</str>

Adding a Button to the Microsoft Outlook Ribbon: Worked Example

This example demonstrates how to add an Oracle Fusion CRM for Microsoft Outlook button to the Microsoft Outlook 2007 and 2010 Ribbon.

Firstly, you add the button to the Microsoft Outlook Ribbon, by editing either the toolbars_12.xml or the toolbars_14.xml client configuration file. Then you register the button, by editing the application_script.js client configuration file. Lastly, you define what action the button will execute when the button is clicked, by editing the actions.js client configuration file.

Editing the toolbars_12.xml, or the toolbars_14.xml Client Configuration Files

1. Use an XML editor to open either the toolbars_12.xml client configuration file (for Microsoft Outlook 2007), or the toolbars_14.xml file client configuration file (for Microsoft Outlook 2010).

2. Find the following section in the client configuration file:

<custom_ui for="Microsoft.Outlook.Contact">
... 
<tabs>
<table idMso="TabContact">
<group id="FsnInspector2" label="#toolbar_actions"
3. Below the section specified in step two, enter the following code:

```html
<button id="button_id" label="#resource_to_display" size="size"
        image="resource_for_icon"
        getVisible="get_visible" getEnabled="get_enabled"
        onAction="button_on_action"/>
```

Substitute the values within the quotation marks with your button information, as follows:

- **button_id** - enter the button’s unique ID
- **resource_to_display** - enter the button label
- **size** - enter the button’s display size. This can be either small, normal, or large.
- **resource_for_icon** - enter the URL for the button icon

**Editing the application_script.js Client Configuration File**

1. Use a JavaScript editor to open the application_script.js client configuration file.

2. Go to the end of the application_script.js file, and register the new button using the following code:

```javascript
action_manager.add_action("button_id", new actions.some_action,
                        helpers.merge_contexts(toolbar_options, options));
```

Substitute the following values with your button information:

- **button_id** - enter the ID of the button you want to register
- **actions.some_action** - the action you will define in the actions.js client configuration file
- **options** - options for configuring action control.

The options that you can specify are as follows:

```javascript
{
    "default": <"hidden"> | <"enabled"> | <"disabled"> | <"type_dependent"> | 
    <"type_dependent_disabled">,
    "no_selected": <"hidden"> | <"enabled"> | <"disabled"> | 
    <"type_dependent"> | <"type_dependent_disabled">,
    "single_selected": <"hidden"> | <"enabled"> | <"disabled"> | 
    <"type_dependent"> | <"type_dependent_disabled">,
    "multi_selected": <"hidden"> | <"enabled"> | <"disabled"> | 
    <"type_dependent"> | <"type_dependent_disabled">,
    "type_dependence": { <"type_id"> : <"hidden"> | <"enabled"> | 
                        <"disabled"> | <"context_dependent_enabled">, ... },
    "all_types_state": <"hidden"> | <"enabled"> | <"disabled">
}
```

The descriptions of these options are as follows:

- **no_selected** - state of the action control when no items are selected
• **single_selected** - state of the action control when a single item is selected
• **multi_selected** - state of the action control when multiple items are selected
• **type_dependence** - state of items by their types (if the state is **type_dependent**)
• **all_types_state** - state of items by their type, that is not defined in the **type_dependence** option

The possible states of the options are:
• **hidden** - action control is hidden
• **enabled** - action control is enabled
• **disabled** - action control is disabled
• **type_dependent** - state of the action control depends on the selected item type
• **type_dependent_disabled** - state of the action control depends on the selected item type, but the control state is **disabled** by default
• **context_dependent_enabled** - state of the action control depends on the context

**Editing the actions.js Client Configuration File**

1. Use a JavaScript editor to open the action.js client configuration file.
2. Define what action will occur when the button is clicked, by entering the following code:

```javascript
function some_action(ctx, options) {
    this.execute = function (action_ctx) {} 
    this.is_enabled = function (action_ctx) {} 
}
```

Substitute the following values with your button action information, as follows:
• **execute** - a function that is executed by a button click
• **is_enabled** - a function that returns a boolean value, that determines whether the button is enabled

**Displaying a custom Oracle Fusion field in a Oracle Fusion CRM for Microsoft Outlook form: Worked Example**

This example demonstrates how to display a custom Oracle Fusion field in a Oracle Fusion CRM for Microsoft Outlook form. This example specifically shows you how to display the Opportunity Number field on the Opportunity form.
Firstly, you need to make the field Opportunity Number available through the Oracle Fusion API, and then customize CRM desktop to synchronize and display the field.

**Define the field to synchronize**

1. Open the MetaInfo.xml file using an XML editor.
2. Find the following tag:

   `<Type Key="OptyId" Label="#obj_opportunity"...>`

3. Add the following field tag as a child to this object:

   `<Field FieldName="OptyNumber" FieldDef="xsd:string"></Field>`

**Define the field on Opportunity object type**

1. Open the fusion_basic_mapping.xml file using an XML editor.
2. Find the following tag:

   `<type id="opportunity" display_name="#obj_opportunity_plural" folder_type="10">`

3. Map OptyNumber to Opportunity type. To do this, add the following snippet as a child to this Opportunity type:

   ```xml
   <field id="OptyNumber">
     <reader>
       <mapi_user>
         <user_field id="fsn Opportunity Number" ol_field_type="1"></user_field>
       </mapi_user>
       <convertor>
         <string/>
       </convertor>
     </reader>
     <writer>
       <outlook_user>
         <user_field id="fsn Opportunity Number" ol_field_type="1"></user_field>
       </outlook_user>
     </writer>
   </field>
   ```

4. To map the display name, open the package_res.xml file and add the following code below the `<res_root>` section.

   ```xml
   <str key="lbl_opty_number">Opportunity Number:</str>
   ```

**Insert a label and the OptyNumber field**

1. Open the forms_12.xml file using an XML editor.
2. Insert the label and the OptyNumber field below the comment <!-- opportunity number --> on the Opportunity form.

   ```xml
   <!-- opportunity number -->
   <cell size="20">
   ```
3. Find the following comment and the cell containing a list of controls
   extended in the previous step and resize its value to 220.

   <!-- opportunity details -->
   <cell size="197">

   **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
- 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. There are three types of customers: Organization, Person, and Group. 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.

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.

---

**Manage Customer Center Trees**

**Customer Center Trees: Explained**

There are two types of Oracle Fusion Customer Center trees. Each tree displays a different set of nodes based on party type. 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 two types of Customer Center trees are:

- Customer Tree
- Contact Tree

**Customer Tree**

The customer tree displays similar nodes for the three customer types, organization, person, and group, such as Profile, Contacts, Sales Account Team, or Assessment. 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, and notes. 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.

**Contact Tree**

The contact tree displays nodes for contacts with the contact profile and other related information such as the Profile (contact details), interactions with the contact, notes, and so on. Leads and Opportunity applications can be accessed here. All nodes on the contract tree are visible to all users.

---

**Manage Customer Center Trees: Explained**

Customer center tree is a navigation paradigm which enables quick and easy access to various related information in one central place. Seen on the regional area of the page, the tree is made up of object nodes such as Profile or Contacts. These object nodes can be categorized into logical categories. Categories enable you to organize those object nodes to fit your needs, for example, the Sales category or Service category. Each implementation can customize the Customer Center tree by showing or hiding the various nodes as required, and configuring node names and other parameters. When saved, the personalizations for this view of the tree are kept for all users of the application. Individual users will have capability to further personalize the tree as desired.
Managing Customer Center Trees

Set these attributes for each node in the Customer Center Tree:

- **Name** - the name shown in the customer center tree UI.
- **Visible** - indicates whether the node will be visible in the customer center tree.

**Important**

All tree nodes that render portlets are delivered with the Visible check box unselected. To show the portlet, select the Visible check box.

- **Default** - the node shown when a user drills down into the customer center tree.
- **Portlet** - indicates whether the node is a portlet or a local task flow. A portlet is a non-local task flow residing in another business process. For example, when accessing the Opportunities node in the Lead Management application, the Opportunities node is a portlet because the Opportunities task flow resides in the Sales applications, outside of the local Lead Management application. Each Fusion application using Customer Center is delivered with the appropriate portal information already configured and should not be changed. All tree nodes that render portlets have the 'Visible' flag turned off. If the portlet is required to be visible, the 'Visible' flag needs to be changed to show the node.
- **Parameters** - specify input variables and values for the node. There are only three nodes that require parameters. These nodes are specifically for third-party integration: OneSource Profile, Service Requests, and Snapshot:
  - **OneSource Profile parameters**: `token=#{'{OneSource token}'}`
    
    Replace `{OneSource token}` with your OneSource access token. For example, if your OneSource token is 'token', set the OneSource Profile parameter as: `token=#{'token'}`. Or, if you do not require a token to access OneSource, simply replace `{OneSource token}` with `NULL`, set the OneSource Profile parameter as: `Token=#{''}`
  
  - **Service Requests parameters**: `HostName=#{'{Siebel server path}'}; SSLEnabled=#{'{true|false}'}; UserName=#{'{username}'}; Password=#{'{password}'}; System Name=#{'{reference system name}'}

    1. Set host name to be your Siebel server path, for example, `HostName=#{'hostname.siebel.com/ CALLCENTER_enu/start.swe'}`
    2. Set SSLEnabled to true or false, for example, `SSLEnabled=#{'false'}`.
    3. Set UserName to be your Siebel system login, for example, `UserName=#{'USER'}`.
    4. Set Password to be your Siebel system password, for example, `Password=#{'PWD'}`.
5. Set System Name to be your source system name as defined in the Original System References mapping table, for example, System Name=#{'SIEBEL'}. The default value is 'SIEBEL' if this parameter is not specified.

Example Service Requests parameter:
HostName=#{'hostname.siebel.com/CALLCENTER_enu/start.swe'};SSLEnabled=#{'false'};UserName=#{'USER'};Password=#{'PWD'};System Name=#{'SIEBEL'}

• **Snapshot node parameter**: HostName=#{'{Siebel server path}'};SSLEnabled=#{'[true|false]'};UserName=#{'{username}'};Password=#{'{password}'};System Name=#{'{reference system name'}

The Snapshot node parameter is the same as the Service Requests node and thus needs the same parameters as those for the Service Requests node.

**Third-Party Integrations in Customer Center: Explained**

Oracle Fusion Customer Center is a central location to access a comprehensive and multifaceted view of customer information. It unifies Fusion applications data as well as relevant third-party content.

**Third-Party Integrations in Customer Center**

OneSource and Siebel Service are two third-party integrations readily configured in Customer Center. This topic explains how third-party customer content is mapped to Fusion customer.

• **OneSource to Fusion Mapping**

OneSource, an online source of business and company data, can be accessed directly from the OneSource node in Customer Center.

Customer Center conducts searches for OneSource company data in the following order:

a. Look up based on mappings defined in HZ_ORIG_SYS_REFERENCES table where orig_system is ONESOURCE.

b. Look up based on Fusion customer stock symbol. This is checked if mapping is not found in HZ_ORG_SYS_REFERENCES.

c. Look up based on Fusion customer name. This is checked if mapping is not found by stock symbol lookup. If there are multiple OneSource companies match the Fusion customer name, user can choose from the list of matching OneSource companies.

• **Siebel Service to Fusion Mapping**

Mappings for Siebel accounts to Fusion customers are maintained in the HZ_ORIG_SYS_REFERENCES table, where orig_system is SIEBEL.
Note

Customer Center does not include licenses for OneSource and Siebel applications. Third-party application licenses may be acquired separately. If you want to enable OneSource and you have a web proxy for external HTTP(S) traffic, you must select Enable Web Proxy on the Web Proxy Configuration screen and specify your web proxy configuration.

See Also: "Web Proxy Configuration" in the chapter "Creating a New Provisioning Plan" of the Oracle Fusion Applications Installation Guide

FAQs for Manage Customer Center Trees

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 be the default node that will display upon opening the tree. When you save, the customization will be associated to your user name.

Manage Customer Center Assessment Template

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.
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 normalized 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.
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.

---

**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
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 Manage Customer Center Assessment Template**

**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.

**Manage Customer Center Task Template**

**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.

**Resulting Task Template**

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 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.

FAQs for Manage Customer Center Task Template

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 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.
Enhanced Click-to-Dial Components: How They Work Together

Oracle Fusion Click-to-Dial is a user friendly way to receive calls from your customers and contacts and to record those interactions. Outgoing calls are made through the hyper-linked phone numbers seen displayed with customer and contact information and through contextual action dialogs. A Click-to-Dial tool bar is used as a dashboard for call status and taking actions on inbound and outbound calls. The tool bar is supported by an infrastructure of connectors and telephony systems that enable this to happen. A delivered generic application program interface (API) enables Click-to-Dial to work with any switch or telephony vendor. Oracle does not author or maintain these connectors, except for Oracle Contact Center On Demand (COD), which is also a reference implementation. For this release, the COD connector is supported only on Windows desktops and Linux. This restriction will be removed in later releases.
Agents

An agent initiates their availability for inbound voice communications by signing in to a connector via the tool bar. The agent signs in via a single connector for that toolbar session. Each connector is associated with an agent or agents. The agent has an assigned agent group, agent identifier and password for each connector. This enables the Agent to sign on to a connector and use the tool bar to receive and transfer calls to other agents and customers or contacts. There may also be other parameters required at this agent level for the telephony system. These can be designated at the agent connector parameter level and passed by the connector to the telephony system. Agents may also record each of these transactions via the Oracle Fusion Interaction features.

Connectors

Connectors perform the linking of the agent to the telephony system that enables the outgoing and incoming calls via Click-to-Dial. The Oracle Contact on Demand Connector is delivered. Contact On Demand is a multichannel, all-in-one call center application that itself connects to the Public Switched Telephony Network (PSTN) to provide voice communication capability. Other connectors may be added using the delivered template.

Telephony Systems

These are third party systems used to establish telephone communications to and from the customer site. They can operate over the internet protocol layer or on hard wired telephony networks.

Tool Bar

This user interface to the Click to Dial feature enables you to establish the connection with the software connector to the telephony system. You may receive, ignore, hold, or transfer a call from your customers and contacts. You can also consult with another agent with the calling party on the line or on hold.

The Oracle Fusion Interactions application can be used to record the particulars of the call using the interactions user interface. Upon receiving a call the tool bar will use the incoming phone number to identify the caller. If the caller is known, an interaction window will be created and opened. If the caller is unknown, the tool bar will popup a window allowing you to search the customers or contacts available to you. If the company is known but not the caller, the list of contacts available to you from that company will be searched. You can also manually open a search dialog. Once the call is completed the Interaction wrap-up screen can be programmed to popup at the end of the call enabling the Agent to finish noting any further call details.
Assignment Object Components: How They Work Together

Work objects, candidate objects, and attributes are components used to create the assignment objects that are used in rule and territory-based assignment.

When you create work objects and candidate objects, you can select attributes which will be used in assignment mappings or rules. For example, the lead work object attribute Primary Product can be added as an assignment attribute and later used in a mapping to associate it with the territory (candidate object) product dimension.

Candidate objects are also available to associate with work objects. For example, the Territory candidate object is associated with the Lead work object. This is used to drive the assignment of territories to a lead.
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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the object with an optional description.</td>
</tr>
<tr>
<td>Code</td>
<td>A unique code used in processing the object.</td>
</tr>
<tr>
<td>Work/Candidate Object check boxes</td>
<td>Indicates if the object is a work object, candidate object or both.</td>
</tr>
<tr>
<td>Application Module</td>
<td>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.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Module configuration</td>
<td>Valid for Top Level Work and Candidate objects except Classification Candidate objects. Child objects will automatically inherit this value from its parent.</td>
</tr>
</tbody>
</table>
| View Object Instance              | Used to define the data model of a view object component when designing an application module, for example, Lead. 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. 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 an Assignment Attribute

Attributes are elements in the 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 Industry, Customer Size, and Organization Type. When you configure assignment mappings for the sales account work object, your chosen attributes are available. You could create a mapping for Sales Account using the Industry attribute.

Having selected an attribute, you can also select the diagnostic display attribute that will be shown when assignment is run in diagnostic mode. For example, selecting the address attribute which represents the user display field associated with the geography identifier attribute.

When selecting attributes for a candidate object, you will want to select the attributes you want to use when configuring assignment rules and mappings that involve that candidate object. 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 specify the order in which these attributes appear in the recommended candidates screen.

Relating Candidate Objects and Work Objects

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.

To relate a candidate object to a work object, use the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assign Candidates</td>
<td>Indicates that Assignment Manager performs the assignment. If not set, then Assignment Manager is used to find the matching candidates, which are then passed to the calling application to update the work object. The behavior is seeded for each object being assigned and cannot be changed by implementations.</td>
</tr>
<tr>
<td>Custom Logic</td>
<td>Indicates that Assignment Manager passes the result of the assignment matching to the callback function of the work object. For example, Opportunity Management uses custom logic that updates the sales team with the territory members. It stamps the territories onto the revenue line and adds the territory team members (resources) to the opportunity sales team. This logic cannot be changed by implementations.</td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Merge Assignment Candidates</td>
<td>Controls whether the matching assignment candidates identified from processing each set of mappings should be merged. This is used to drive the merging of matching candidates when multiple mapping sets are used in assignment processing. If the check box is checked, then the candidates are merged. The default is unchecked and should not be changed unless under the guidance of Oracle Support.</td>
</tr>
<tr>
<td>Keep Manual Candidates</td>
<td>Indicates that manually assigned candidates are retained during assignment processing. This option can be used to prevent the removal of manually added candidates during reassignment. Sales Account and Opportunity Management have implemented their own lock assignment features and do not leverage this option.</td>
</tr>
<tr>
<td>Replace Candidates</td>
<td>Determines whether unqualified candidates are removed from a team when an assignment runs. For example, the first time that Assignment Manager runs, a territory is assigned to a sales lead. When the reassignment process runs following a territory proposal activation, the territory is no longer valid. If Replace Candidates is set, then the territory is removed from the sales lead.</td>
</tr>
<tr>
<td>Candidate Exclusion</td>
<td>Sales leads have a related object, which stores the excluded candidates for each sales lead. Assignment Manager accesses this information, which prevents assignment of the work object to an excluded candidate. This option is only relevant to Sales leads.</td>
</tr>
<tr>
<td>Parent Attribute</td>
<td>Used by territory-based assignment to determine the hierarchy of matching territories, eliminate all parent territories, and only return and assign the matching leaf node Territories. If this attribute is not used, then all matching territories (parent or leaf) are returned and assigned.</td>
</tr>
<tr>
<td>Candidate Differentiation Attribute</td>
<td>Stores the attribute on the candidate object that is used for discriminating matching candidates. For example, for a work object like lead, you might choose the attributes of Primary Product, Customer Geography, Customer Industry, Customer Size, and Organization Type. When you configure assignment mappings for the lead work object, your chosen attributes are available. You could create a mapping for Lead using the Primary Product attribute. The setting of this attribute should not be changed unless under the guidance of Oracle Support.</td>
</tr>
<tr>
<td>Coverage Attribute</td>
<td>The territory attribute used to denote whether the candidate in the matching candidate list has a regular, included, or excluded coverage.</td>
</tr>
<tr>
<td>Maximum Number of Candidates</td>
<td>The maximum number of candidates returned for the work object and candidate object combination. The default value is 100. Implementations may need to change this value if a large number of candidates can be assigned.</td>
</tr>
<tr>
<td>Manual Attributes</td>
<td>The attribute that identifies a candidate was manually assigned, rather than by the system. This attribute is used with the Keep Manual Candidates attribute.</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Keep Manual Candidates</td>
<td>A flag to retain manually assigned candidates when assigning or re-assigning a work object. This option is only relevant if the manual attribute is defined and the Assigned Candidates option is checked.</td>
</tr>
<tr>
<td>Replace Candidates:</td>
<td>Indicates whether non matching candidates will be removed when re-assigning a work object. In an example scenario, in the first time assignment runs, territory A is assigned to a sales lead and there is a change within the territory definition. When the sales lead is reassigned, territory A is no longer valid. If this option was selected, then territory A is removed from the sales lead.</td>
</tr>
<tr>
<td>Score Attribute</td>
<td>The score attribute in which the calculated score is stored.</td>
</tr>
<tr>
<td>System Attribute</td>
<td>The attribute that identifies a candidate was assigned by the system, rather than manually.</td>
</tr>
</tbody>
</table>

**Setting Up Assignment Mappings**

Assignment mapping sets and their related mappings drive territory-based assignment. The mapping sets determine which mappings are used and the sequence in which mapping sets are used in territory-based assignment. The mappings identify the dimensions, attributes, and territory filtering used in the assignment processing. Default mapping sets and their related mappings are seeded.

**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, and to create the rules and mappings that dictate the selection and assignment of resources and territories. 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
- Attributes
- Resources and territories
- Assignment disposition
- Mappings sets and mappings
• 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 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.

A set of business or assignment objects is seeded for the assignment of territories or resources to sales accounts, partner accounts, opportunities, and leads.

Attributes

To ensure that candidates are properly assigned to work objects, 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, 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 specific sales representative to a business object, such as opportunity, based on the risk level of the opportunity. In this case, you will select the attribute of the opportunity work object that corresponds with risk level, and the attribute of the resource candidate object that corresponds with the name or E-mail address. 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 candidate objects to work objects.

Mappings Sets and Mappings

Assignment mapping sets and their related mappings drive territory-based assignment. The mapping sets determine which mappings are used, and the sequence mapping sets are used in territory-based assignment. The mappings identify the dimensions, attributes, and territory filtering used in the assignment processing. Default mapping sets and their related mappings are seeded.

Rules

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.

Create the rules using the work objects, candidate objects, and attributes that you already established. When designing your rules, carefully consider how you want to match candidates to work objects. For example, would you want resources assigned based on their geographic location, their product knowledge, on the status or score of an object, 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 rules.
Assignment mapping sets and their underlying mappings drive territory-based assignment. This topic explains how these components work together in assignment processing.

The mapping sets determine which mappings are used and their sequence of use in territory-based assignment. Mapping sets allow different groups of attributes or dimensions to be used when matching territories.

Mappings

The mappings identify the dimensions, attributes, and territory filtering used in the assignment processing. Default mapping sets and their related mappings are predefined for sales account, leads, partner accounts, and opportunity revenue assignment. This predefined mapping assumes that opportunities, leads, sales, and partner accounts use the same territory hierarchy.

Each predefined mapping set has between 9 and 16 mappings that determine the information on the object, such as the sales account industry or the sales lead product, and how each is mapped to a dimension or attribute on the territory.

You can create additional mappings using the work objects, candidate objects, and attributes that you already established.

Mapping Sets

Mapping sets enable the grouping of mappings so that you can create more than one mapping for each combination of work object and candidate object. The mapping set concept is used only with territory-based assignment and territory-based assignment with rule filtering. Mappings sets are predefined for sales accounts, leads, opportunities, and partner accounts. When managing assignment objects, the user can define additional mapping sets, each of which is comprised of multiple mappings, for each combination or work object and candidate object.

Mapping Types

There are three types of assignment mapping:
**Dimension Mapping:** Dimension mappings 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 is passed to the translation function, in case the same function is used for multiple dimensions.

When creating the mapping, the Function Service and Function Code are only needed if a translations function is used. The function code field is used to specify a unique identifier for the attribute, and this identifier is passed to the translation function.

An example is assigning territories to opportunity revenue lines based on the product associated with the revenue line. In this case, dimension is selected as the mapping type. The candidate object low attribute and high attribute correspond to the names of the low sequence and high sequence attributes for product on the territory. The work object low attribute and high attribute correspond to the names of the low sequence and high sequence attributes for product on the revenue line.

**Attribute Mapping:** This mapping enables you to compare and match attribute values between a work object attribute and a candidate object attribute. When the value of the candidate object 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 Partner Identifier attribute and the territory object with Partner ID attribute. The selection criterion is: select Sales Lead Territories where Sales Lead Territory.Partner Identifier equals Sales Lead.Lead Partner Identifier. The assignment engine will use this mapping data to construct a query on the candidate object that is equivalent to the selection criteria.

**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 Coverage Model that equals the value PARTNER_CENTRIC.

---

**Note**

For Literal Mappings, ensure that the value entered corresponds to the Lookup Type Value code, not the meaning.

---

**Assignment Processing using Mapping Sets and Mappings**

When designing your mappings, carefully consider the dimensions and attributes you use in your territory structure and how you want to match these territory candidates to work objects. Also consider the shape of the information used in the territory structure; this may affect the sequence of each mapping. A sequence can be entered for each mapping set which is used to determine the order in which these mapping sets will be used in the territory-based assignment processing. The sequence of the dimension mappings used in territory matching can affect performance. The most selective dimension mapping should be given lowest sequence number. By default, this dimension is the Geography Dimension. By using the lowest sequence number, it is performed earliest in the matching process, which results in the smallest number of territory matches.
Mappings that do not have a sequence are used together at the end of the matching process.

Sometimes the mapping set sequence does not matter, for example, there are two predefined opportunity revenue assignment mapping sets. When the first mapping set is used, it finds matching territories based on the information on the opportunity/opportunity sales account, and the territory information. Then the second mapping set is used which matches territories based on the opportunity/opportunity partner information and the territory information. The order of the mapping sets are interchangeable; regardless of which mapping set is used first, the resulting territories that match will be the same.

In the case of leads, the mapping set sequence is important as the territories matched using the first mapping set may result in a primary partner being added to the lead. This information is significant to the territory matching performed using the second mapping set.

Mapping sets can be made conditional to control whether the mapping set is used or not used during assignment processing. For example, the partner channel manager territory assignment mapping set conditional attribute is set to the value RevenuePartnerId. During the assignment processing of a revenue line, if the Revenue PartnerId attribute for that revenue line contains a value, then this mapping set will be used in territory matching processing.

Assignment Manager has an indicator in the related Candidates region to control whether to merge the matching assignment candidates identified from processing each set of mappings. This indicator is used to drive the merging of matching candidates when multiple mapping sets are used in assignment processing. If the box is checked, then the candidates are merged. The default is unchecked.

In most implementations, the predefined mapping sets will be sufficient, but mapping sets can offer some flexibility if custom assignment processing is needed.

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 partner ID is the same as the sales lead partner 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 partner ID is equal to the sales lead attribute partner ID. Enter a value for the sequence which determines the order in which the mapping is used when matching territories. The most selective mapping should be given lowest sequence number, the next most selective mapping should be given the next sequence number.
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 dimension as the mapping type and enter the value **Prod** for the function code. Enter a value for the sequence which determines the order in which the mapping is used when matching territories. The most selective dimension mapping should be given lowest sequence number. 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 **InventoryItemId**.

- 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 **ProdGrpId**.

- 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 have a sales account centric coverage model. For example, territory coverage model equals **SALES_ACCOUNT_CENTRIC**.

Manage Assignment Manager Rules

Rule Set Components: How They Work Together

The rule set type, filter settings, and rule action are rule set components that work together to tell the assignment manager engine how to process rule-based assignments for work objects.

A rule set type is set at the rule set level; and two of the rule set types, Matching Candidate with Scoring and Matching Candidate, require additional filter settings. At the rule level within a rule set, an action setting is entered that
determines the action that is performed when a rule is evaluated as true. The rule action works in conjunction with the rule set type.

The rule administration feature prevents more than one user at a time from updating assignment rules for a product. For example, if User A is currently updating assignment rules, and User B also attempts to update assignment rules for that application at the same time, the changes made by User B will not be saved, and an error message will appear.

In the case of Assignment Rule Administration, only one user should edit Sales Lead Assignment Rules at a time. The same applies to Opportunity Management Assignment Manager Rules.

**Rule Set Type**

The rule set type on the rule set determines the type of rule-based assignment processing to be performed. For example, when the rule set type is Matching Candidate, the candidates that match the conditions of the rules evaluated as true by the assignment engine are assigned to the work object. The number of matching candidates that are assigned to the work object is determined by the rule set filter settings.

**Filter Settings**

The filter settings are used in conjunction with two of the rule set types: Matching Candidate and Matching Candidate with Scoring. The filters allow
you to indicate how many matching candidates you want to assign to the work object. When set to All Above Minimum Score, all of the matching candidates above a particular score are assigned to the work object. Set the score in the Minimum Score field.

When set to Top X, a number of matching candidates with the highest scores are assigned to the work object. Use the Number of Candidates field to specify how many top matching candidates to assign.

When the rule set type is Matching Candidate, and the filter is set to Random, a random selection of matching candidates are assigned to the work object. When the rule set type is Matching Candidate with Scoring, and the filter is set to Random, a random selection of matching candidates with the highest scores is assigned to the work object. Use the Number of Candidates field to specify how many random matching candidates to assign.

**Action**

The action setting determines the action that is performed when a rule is evaluated as true. The action setting is the one component that is set at the rule level rather than the rule set level; however, it does work in conjunction with the rule set type. When the rule set type is Classification, the rule action can only be Return the Candidate value as <value>. For example, the work object for a rule set is Lead, and the candidate object is a classification object called Lead Qualification. The rule set type is set to Classification, and the action for one of the rules in that set is Return the candidate value as QUALIFIED. If that rule is evaluated as true, the Lead Status for the Lead being classified is set to Qualified.

When the rule set type is Matching Candidate, the rule action can only be Return matching candidates. If a rule with that action is evaluated as true, the candidates that match the conditions for that rule are assigned. The filter setting at the rule set level determines whether all matching candidates are assigned (All), or a random number of matching candidates are assigned (Random).

When the rule set type is Matching Candidate with Scoring, the rule action can only be Increase the matching candidate score by <value>. If a rule with that action is evaluated as true, the candidates that match the conditions for that rule get the value in the action added to their score. For example, the work object for a rule set is Opportunity, and the candidate object is Resources. The rule set type is set to Matching Candidate with Scoring, and the action for one of the rules in that set is Increase the matching candidate score by 10. If that rule is evaluated as true, the resources that match the conditions for that rule get 10 added to their scores. The scores are cumulative, so if any of the resources that matched the conditions in the rule in the example also match the conditions for other true rules in the set, those territories get additional values added to their current score of 10. The filter setting at the rule set level determines whether all matching candidates are assigned (All), or all matching candidates above a specified score are assigned (All Above Minimum Score), or a random selection of matching candidates with the highest scores are assigned (Random), or a number of matching candidates with the highest scores are assigned (Top X).

When the rule set type is Scoring, the rule action can only be Increase the score by <value>. If a rule with that action is evaluated as true, the value in the action is added to the score of the work object associated with the rule set. For example, the work object for a rule set is Lead. The rule set type is set to Scoring, and the action for one of the rules in that set is Increase the score by 20. If that rule is evaluated as true, the score for the Lead is increased by 20.
Creating Assignment Rules: Examples

Assignment rules are created using work objects, candidate objects, attributes, conditions and actions. The assignment engine uses your rules to evaluate and recommend candidate assignments for specified work objects. There are multiple types of rules you can create. The following scenarios illustrate each type:

Managing Classification Rules
When the following attributes for leads are set as specified, you want to classify those leads as qualified:

- Lead Customer is a sales account
- Lead Product is not NULL
- Lead Score is greater than 150

Create a rule set with a rule set type of Classification Rule. Set the work object as lead and the candidate object as lead qualification. Create a rule with the three conditions that match the attribute settings you want a lead to have in order to be considered an qualified lead. Using the Lead Product condition as an example, you would choose the Lead attribute name **Primary Product ID**, and then select the Does Not Equal operator. Finally, enter the value of NULL. For the Lead Customer condition, you would choose the Lead attribute name **Sales Account Indicator**, and then select the Equals operator. Finally, enter the value of Y. For the Lead Score condition, you would choose the Lead attribute name **Score**, and then select the Greater Than operator. Finally, enter the value of 150. Enter the action for your rule is Return the Candidate Value As **Qualified**.

Managing Scoring Rules
When the following attribute for leads are set as specified, you want to increase those leads’ scores by 150:

- Lead Time Frame is 3 months

When the following attributes for leads are set as specified, you want to increase those leads’ scores by 100:

- Budget Status is Approved
- Budget Amount is > 500000

Create a rule set with a rule set type of Scoring Rule. Set the work object as lead, and create the first rule with one condition that matches the attribute settings you want a lead to have in order to add 150 to its score. Using the Lead Time Frame condition as an example, you would choose the attribute name Time Frame, and then select the Equals operator. Select the value 3 Months. Then enter the action for your rule as Increase the Score by 150. Create your remaining rule with two conditions and action to Increase the Score by 100.

Managing Matching Candidate Rules
Identify a single candidate territory for your sales leads in one line of business and all territories for the other. Create a rule set with a rule set type of Matching Candidate Rule, a filter type of Random, and Number of Candidates of 1. Set the work object as sales lead and the candidate object as sales lead territory, and create a rule with two conditions. The first condition is the response channel on
the Sales Lead equals LOB1, and the second condition is the territory type equals ignoring case Partner. If this rule is true, then only a single random matching partner territory will be assigned to the Sales Lead.

For the second line of business, create a rule set with a rule set type of Matching Candidate Rule, and a filter type of All. Set the work object as sales lead and the candidate object as sales lead territory. Create a rule with one condition with the response channel on the Sales Lead equals LOB2. If this rule is true, then all matching territories will be assigned to the Sales Lead.

Managing Matching Candidate with Scoring Rules

Assign different country specialists to opportunities in some European countries based on the country and the risk level of the Opportunity.

Create a rule set with a rule set type of Matching Candidate with Scoring Rule, a filter type of All Above Minimum Score, and the minimum score set to 20. Set the work object as opportunity and the candidate object as resource, and create two rules each with two conditions.

The first rule has two conditions. First, choose the work object of Opportunity, choose the attribute name Customer Country, select the Equals operator, and then enter the value DE. For the second condition, choose the work object of Resource, and then choose the attribute name Name, select the Equals (=) operator, and then enter a value of John Brooks. Select the action for the rule as Increase the Matching Candidate Score by 20.

The second rule has two conditions. First, choose the work object of Opportunity, choose the attribute name Country, select the In operator, and then enter the values FR and UK. For the second condition, choose the work object of Resource, choose the attribute name Name, select the Equals (=) operator, and then enter a value of Claire Stevens. Select the action for the rule as Increase the Matching Candidate Score by 20.

The second rule has three conditions. First, choose the work object of Opportunity, choose the attribute name Risk Level, select the Equals (=) operator, and then select a value of High. For the second condition, choose the work object of Resource, choose the attribute name Name, select the Equals (=) operator, and then enter a value of Claire Stevens. Select the action for the rule as Increase the Matching Candidate Score by 20.

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 matching resources. In Oracle Fusion Partner Relationship Management implementations, all territories (prime, overlay, partner, and so on) matching a given lead may be identified. Rule filtering may 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 and 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. If the deal is internal then only
the prime territories are assigned. If the deal is pushed to a partner, a channel manager is also assigned to oversee the deal.

**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 Assignment Manager 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. Assignment processing then calls the Rule Set Group that contains the rules for the territory-based assignment with rule filtering.

While the territory-based assignment delivers a list of territories, the rules can fine tune the assignment process:

1. **Rule for SALES CHANNEL Does Not Equal NULL**
   a. SalesLead.Sales Channel Does Not Equal NULL
   b. Action: Return matching candidates

2. **Rule for SALES CHANNEL Equals NULL, Assign Channel Manager**
   a. Sales Lead.Sales Channel Equals NULL
   b. Sales Deal.Deal Size Greater Than 1,000,000
   c. Territory.Territory Type In Partner, Sales Channel Manager
   d. Action: Return matching candidates

3. **Rule for SALES CHANNEL Equals NULL, Assign Prime**
   a. Sales Lead.Sales Channel Equals NULL
   b. Sales Deal.Deal Size Greater Than 1,000,000
   c. Territory.Territory Type Equals Prime
   d. Action: Return matching candidates

**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 if this is a smaller deal that can be handled by your partners (and a Sales Channel Manager to oversee), or it is a larger deal that needs to be followed up by the internal sales force.

**Resulting Assignments**
The assignment engine first identifies the 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 it does, then all territories identified (by way of territory-based assignment) are assigned.

2. The second rule says if there is no sales channel assigned, and the deal is under one million dollars, assign the Lead to a Partner and Sales Channel Manager.
3. And the final rule is used when there is no Sales Channel value, and the deal is greater than one million dollars, the lead is assigned to the prime (internal) territories.

FAQs for Manage Assignment Manager Objects

What’s the difference between rule-based and territory-based assignment?

Territory-based assignment is the primary means of assignment for CRM objects. For territory-based assignment, you create work object to candidate object mapping sets during assignment object creation that are used to make candidate assignments.

Rule-based assignment is used for identifying supplemental resources or filtering the matching territories. Rules can also be used for scoring work objects and also for classification. For rule-based assignment, you use the rules editor to create expression-based rules that the assignment engine uses to make candidate assignments.

Territory plays a fundamental role in Fusion CRM. Using attributes it defines how a company is going to market. That is, how is that company deciding to deploy its Sales Resources to sell to Customers? Every Sales Account, Lead, and Opportunity has a territory or territories assigned. In many cases, customers will implement Territories down to the level of individual Sales Representative (that is, each Sales Representative will have his or her own Territory) in order to enable these capabilities at the granularity that they want, as well as the Sales Planning/Territory Alignment capabilities at the granularity that they want.

In a territory-based assignment scenario, you typically create a mapping set between a work object and a candidate object to make the appropriate candidate assignment. This mapping set can consist of 1 or more mappings between the work object attributes and the candidate object attributes.

A simple example territory-based assignment setup is a single mapping set between the opportunity revenue line work object and the candidate object of territory. This mapping set contains a single mapping which maps the attribute of location on the opportunity (the parent of the revenue line) to the geography attribute on the territory. Any territories where the geography value matches the location of the parent opportunity are matched and assigned to that opportunity revenue line. Another example would be two mapping sets defined between the lead work object and the territory candidate object. The first mapping set determines the assignment of territories to each lead based on a mapping between the industry of the customer on the lead and the territory industry dimension. This would be a literal mapping that filters the territories to those that have a status of finalized and a literal mapping that filters the territories to those that have a sales centric coverage model. The second mapping set is conditional and determines the assignment of partner channel manager territories. This mapping set consists of a mapping between the geography of the primary partner on the lead and the territory geography dimension. This
would be a literal mapping that filters the territories to those that have a status of finalized and a literal mapping that filters the territories to those that have a partner centric coverage model.

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.

The following chart compares the features and pros and cons of territory-based assignment and rule-based assignment:

<table>
<thead>
<tr>
<th>Territory-Based Assignment</th>
<th>Rules-Based Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pros:</strong></td>
<td><strong>Pros:</strong></td>
</tr>
<tr>
<td>• User-friendly territory management</td>
<td>• Familiar concept with expression-based rules</td>
</tr>
<tr>
<td>• Minimal assignment configuration</td>
<td>• Any attribute can drive an assignment</td>
</tr>
<tr>
<td>• Support for hierarchies</td>
<td>• Easy to define and maintain a small number of straightforward rules</td>
</tr>
<tr>
<td>• Preview of assignment</td>
<td>• Scoring and classification</td>
</tr>
<tr>
<td>• Reports to identify gaps and overlaps</td>
<td>Cons:</td>
</tr>
<tr>
<td><strong>Cons:</strong></td>
<td>• Unable to navigate hierarchies</td>
</tr>
<tr>
<td>• Additional administration required even with simple territory models</td>
<td>• Requires intensive effort to manage complex or large numbers of rule</td>
</tr>
<tr>
<td>• Single territory structure for CRM</td>
<td></td>
</tr>
</tbody>
</table>

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 set, 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.
Sales Account and Territory Assignment

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>
<tr>
<td>Auxiliary Dimension 3</td>
<td>the primary classification code for the classification category defined in profile option Industry Classification Category for Auxiliary Dimension 3.</td>
</tr>
</tbody>
</table>

Sales Account Team Member Access Levels: 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**
Assignment Manager

Territory Management must be implemented to utilize this feature.

**Territory Coverage: Explained**

A territory coverage is a set of boundaries that define what is included or excluded in the territory and what can be sold. For example, you can use product and geography dimensions to create a territory coverage for selling laptops in North America.

A coverage includes the following parts:

- **Dimensional Coverage**: The combination of one or more territory dimensions.

- **Inclusion Coverage**: A list of selected customers or partners, regardless of the defined dimensional coverage, if any. The sales accounts for the selected customers will be assigned to the territory. These sales accounts do not have to be designated as Named accounts in Customer Center.

- **Customer Hierarchy**: You can choose to include the customer hierarchy for the selected customer. All sales accounts for the selected customer hierarchy will be assigned to the territory.

- **Filtering Conditions**: Defined dimensions apply to the included customers and their hierarchies so that only sales accounts that match the dimension definitions get assigned to the territory.

- **Other Dimensions**: Product or sales channel dimensions defined for all of the included customers or partners. Product and channel selections must fall within the jurisdiction of the parent territory if you select Restrict by parent in the dimension selection window.

- **Exclusion Coverage**: A list of excluded customers (including related sales accounts) or partners, regardless of the defined dimensional coverage. You can choose to exclude the customer hierarchy for the selected customer. All dimensions are ignored. The sales accounts do not have to be designated as Named accounts in Customer Center to be excluded.

- **Inheritance Dimension Overrides**: If the territory inherits its coverage from another territory, then its coverage exactly matches that of the source territory. But, you can enter override definitions for one or more dimensions. Dimension definitions added to the override supersede the definitions for the same dimensions contained in the dimensional coverage. The overrides are also added to the customer inclusion Filtering Conditions and Other Dimensions tables. Inheritance applies only to account centric territories, not to partner-centric territories.

**Note**

When you activate a territory proposal, a reassignment of sales accounts and opportunity revenue lines occurs for territories affected by changes in dimensional coverages. Changes to customer or partner inclusions and exclusions require a full reassignment process. Leads always require batch reassignment.
Territory Coverages for Partners

A partner is an organization party with a partner profile associated and an assigned Partner usage. Partners are defined in the Partner Center.

Similar to direct sales, channel managers have corresponding sales territories pertaining to partner sales activities. Some channel managers are assigned to specific partners. Some channel managers are assigned to customers for sales activities that involve partners. Channel manager territories can be defined by the following coverage models.

- Coverage Defined by End Customer Characteristics
  The territory of the channel manager is solely defined based on the characteristics of the end customer, irrespective of which partner is associated with the transaction. As an example, a channel manager is assigned to cover all the indirect opportunities where the end customer is located in California. You define territories for this account-centric coverage using customer characteristics, and you can include or exclude specific customers.

- Coverage Defined by Partner Characteristics
  The channel manager’s territory is defined based on some characteristics of the partner, such as where the partner is located or the type of the partner (reseller, system integrator, distributor). As an example, a channel manager is assigned to cover all the indirect opportunities where the partner is located in California.

  To define this territory, you designate the coverage model to be Partner Centric instead of Account Centric. In the Partner Centric model, the regular coverage is defined using the following attributes of a partner organization:

  - Primary geographical location of the partner
  - Organization Type of the partner (for example, private, public, government owned, nonprofit)
  - Industries served by the partner (for example, high tech, manufacturing, banking, pharmaceutical)
  - Size of the partner
  - Three auxiliary dimensions are available for partners based on the customer categories classification model

  You can also use the Product and Sales Channel dimensions to match attributes from sales transactions (leads and opportunities).

- Individually Selected Partners
  The coverage for the channel manager is defined using a selection of partners to directly assign to or exclude from his partner-centric territory. As an example, a channel manager is assigned to a partner named AA Solutions. This channel manager will be assigned to all indirect
opportunities where AA Solutions is the partner. The opportunities for included partners can be additionally qualified by product and sales channel.

No Coverage

You can create a territory that has no coverage. The territory is indirectly defined by the coverages of its descendant territories. You can assign quota to the territory and it can participate in forecasting. The territory cannot be automatically assigned to sales accounts, leads, and opportunities, but you can see the assignments of its descendant territories. You can also assign the territory to an opportunity revenue item as an override.

FAQs for Sales Account and Territory Assignment

What's the difference between an internal territory and a partner territory?

An internal, or deploying company, territory is defined, created, and assigned internal resources.

Examples of two internal territories are:

- Sales Representative Territory (SRT) is the jurisdiction of responsibility of a sales representative over a set of sales accounts, leads and opportunities.
- Lead Triage Territory (LTT) is the jurisdiction of responsibility of channel manager to triage partner Leads, that is approve leads and route to the right Partner.

A Partner territory is the jurisdiction of the reselling partner and contains partner resources. Specific Partner territories can be assigned to a sales account as needed.

Note

Territory Management must be implemented to utilize this feature.

How can I add territories to a sales account?

Assignment Manager (part of Oracle Fusion Work Management) is used to determine matching territories for a given sales account. A sales account can also be assigned to one or more internal and partner territories.

All internal territories, such as Prime, Overlay and Sales Channel Manager territories, which match a given sales account’s assignment attributes are assigned to the sales account. Internal territory assignment can be run immediately and automatically whenever sales account assignment or realignment is required. For example, you can run assignment manager when a sales account is created or updated, or when territories are realigned. Internal
territory assignment can also be scheduled to run in a batch, or it can be run on-demand via the Assign Territories action in the sales account team page.

Partner territories are applicable to Partner Relationship Management implementations. When a partner lead is approved, any partner territories associated to the lead are automatically assigned to the lead’s sales account. Channel sales manager can also select specific partner territories to assign to a sales account via the Add Partner Territories action in the sales account team page.

Note
Territory Management must be implemented to utilize this feature.

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.

Manage Sales Lead Assignment Objects

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>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.

**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 XYX company want 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.

**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.

**Batch Mode Automatic Assignment Manager Tasks**

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.

**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.

**Examples of enabling territory dimensions**

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 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 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.

Running Assignment Processes

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.

<table>
<thead>
<tr>
<th>View Criteria Name</th>
<th>View 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>
</tbody>
</table>

[39x39] 6-34 Oracle Fusion Marketing Implementing Marketing
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.

**Candidate Refresh: Explained**

Rule-based assignment requests that identify matching candidates or scores for matching candidates use candidate data such as resources. This candidate data is loaded into a cache and used for each assignment request until the cache is refreshed. Oracle Fusion Assignment Manager supports the ability to refresh
the candidate data cache at regular intervals. The Fusion Enterprise Scheduling Service (ESS) process Refresh Cache is used to set an indicator that the candidate data should be refreshed the next time there is an assignment request using that candidate.

**Note**

This feature affects rule-based assignment using the rule set types of matching candidates or matching candidates with scoring only.

Implementations may schedule this process daily, weekly, and so on, as required by the frequency of changes to the candidates. Consider how often the candidate data will change and how critical it is to have the changes available for use in assignment. For example, resource details may change daily and therefore the resource candidate data cache for Lead Management may need to be updated once per day.

The Refresh Cache process should be defined for each candidate object and application that uses Rules-based Assignment candidate matching or candidate matching with scores. The process has the following parameters:

- Candidate Object Code, for example `Resource_Candidate_Object`.
- Owner Module, such as `sales` or `leadMgmt`.

For example, there would be one ESS process scheduled for Lead Management with the parameter `Resource_Candidate_Object_Lead/leadMgmt`. Sales would need a process scheduled with the parameter `Resource_Candidate_Object/sales`.

### Running Assignment Manager Reports

#### Generating Assignment Reports: Example

Assignment Management allows you to generate the following reports to view batch assignment progress and errors, and to show the sequencing of territory dimensions:

- Batch Assignment Progress Report
- Batch Assignment Error Report
- Territory Dimension Data Report

You must access the Diagnostic Dashboard to run and generate these reports. This topic explains how to run and generate batch assignment progress report, as an example.

**Scenario**

To generate the batch assignment progress report:

1. Log in to the Oracle Fusion application as a user who has access to the Diagnostic Dashboard.
2. Click **Navigator**, and then click **Scheduled Processes** under **Tools**.
3. Run batch assignment processes for **Sales Accounts** and **Opportunities**.
While the process is running, navigate to the Diagnostic Dashboard to run the batch assignment progress report.

4. Click the Help link.

5. Click Troubleshooting, and then click Run Diagnostics Tests.

6. On the Diagnostic Dashboard page, search for the report name you want to run. In this example, select Batch Assignment Progress Report.

7. Select Batch Assignment Progress Report and click Add to Run.

Batch Assignment Progress Report is added under the Choose Tests to Run and Supply Inputs region.

8. Click the warning icon under the Input Status column, and enter the parameters in the Input Parameters page that appears.

9. Click OK.

10. Under the Choose Tests to Run and Supply Inputs region, enter a name in the Run Name field and click Run.

11. In the Confirmation dialog box, click OK.

The status of the report appears under the Diagnostic Test Run Status region.

12. Click the completed report to open the report page.

You can now use the report for your analysis. You can follow the same procedure to generate all reports.

**Batch Assignment Progress Report: Explained**

The batch assignment progress report indicates the number of records processed, unprocessed, successful or failed, and the number of records processed per minute for a process. The report provides details of assignment processing for multiple batch assignment processes and their sub-processes. You can run this report while a batch assignment process (sales accounts, leads, opportunities, revenues, or partner accounts) is running, or after a process has completed.

Access the Diagnostic Dashboard to generate the batch assignment progress report. The report includes two tables, one with details of the main process, followed by details of the sub-processes. The second table with sub-processes appears only if the Include Sub Process parameter is set to True.

**Input Parameters**

The report has the following input parameters:

<table>
<thead>
<tr>
<th>Input Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Process ID</td>
<td>The identifiers of parent ESS processes.</td>
</tr>
<tr>
<td>Note</td>
<td>You can enter multiple process IDs.</td>
</tr>
</tbody>
</table>
Example Report
The table below shows an example of a batch assignment progress report along with a description of what each value means:

<table>
<thead>
<tr>
<th>Column</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Process ID</td>
<td>504</td>
<td>Identifier for the ESS process</td>
</tr>
<tr>
<td>Work Object Code</td>
<td>SalesAccount_Work_Object</td>
<td>The code for the work object</td>
</tr>
<tr>
<td>Candidate Object Code</td>
<td>SalesAccountTerritory_Candidate</td>
<td>The code for the candidate object</td>
</tr>
<tr>
<td>Submitted By</td>
<td>Sales_admin</td>
<td>Submitted By - user name of the person submitting the process</td>
</tr>
<tr>
<td>Process Status</td>
<td>Running</td>
<td>The status of the process, such as, Not Started, In Progress, Cancelled, and so on</td>
</tr>
<tr>
<td>Process Start Time</td>
<td>11/19/12 8:48 PM UTC</td>
<td>Start Time of the process. Shows date, hours, and minutes</td>
</tr>
<tr>
<td>Process End Time</td>
<td></td>
<td>End Time of the process. Shows date, hours, and minutes</td>
</tr>
<tr>
<td>Process Elapsed Time (Minutes)</td>
<td>22</td>
<td>Number of minutes the process has been running</td>
</tr>
<tr>
<td>Records per Minute</td>
<td>10.46</td>
<td>Number of records processed per minute</td>
</tr>
<tr>
<td>Number of Sub Processes</td>
<td>10</td>
<td>The number of sub processes launched from the parent process</td>
</tr>
<tr>
<td>Number of Records</td>
<td>100000</td>
<td>Number of records in the process</td>
</tr>
<tr>
<td>Number Not Processed</td>
<td>94452</td>
<td>Number of records not yet processed</td>
</tr>
<tr>
<td>Number Successful</td>
<td>5542</td>
<td>Number of records successfully processed</td>
</tr>
<tr>
<td>Number Failed</td>
<td>6</td>
<td>Number of records failed</td>
</tr>
<tr>
<td>Failure Rate</td>
<td>0.00006</td>
<td>Number of failures divided by the Number of Items</td>
</tr>
<tr>
<td>Assignment Elapsed Time</td>
<td>21.45</td>
<td>Number of minutes the assignment processing has been running</td>
</tr>
<tr>
<td>Records Processed Last 10 Minutes</td>
<td>12.63</td>
<td>Number of records processed in the previous 10 minutes</td>
</tr>
<tr>
<td>Number of Retries</td>
<td>0</td>
<td>Number of times the process has been resubmitted</td>
</tr>
</tbody>
</table>

The report can be used to estimate the time it will take to complete a batch assignment process. This report provides details on the number of records completed and the number of records in progress. You can generate this report repeatedly to conduct performance analysis of the batch assignment processing.
Batch Assignment Error Report: Explained

The batch assignment error report provides details of the error and warning messages generated while processing individual records during batch assignment process. The report provides a summary of the test input parameters and message details for each record that meets the input parameters. You can run this report while a batch assignment process (sales accounts, leads, opportunities, revenues, or partner accounts) is running, or after a process has completed.

Access the Diagnostic Dashboard to generate the batch assignment error report. The report shows the test parameters followed by two results table. The first table provides a summary of the process, and the second table shows details of the records that meet the criteria entered when running the report.

**Report Parameters**

The report has the following input parameters:

<table>
<thead>
<tr>
<th>Input Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Process ID</td>
<td>Identifier for the parent ESS process.</td>
</tr>
<tr>
<td></td>
<td>Note: You can enter only a single process ID.</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Enter the status of the assignment, such as, error, succeeded, and so on. The default is error.</td>
</tr>
<tr>
<td>Work Object Public Unique Identifier</td>
<td>This is optional. The value that you enter here will depend on the Identifier Attribute of the work object being processed in a batch.</td>
</tr>
<tr>
<td>Range of Records</td>
<td>Enter the range of records in a process to report. The default value is the value set in the MOW_DTF_ERROR_REPORT_MAX_LIMIT profile option. You can change this profile option value in the Manage Administrator Profile Values setup task.</td>
</tr>
</tbody>
</table>

**Example Report**

The following is an example of a batch assignment error report:

**Process Summary**

<table>
<thead>
<tr>
<th>Work Object Code</th>
<th>Candidate Object Code</th>
<th>Start Time</th>
<th>End Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SalesAccount_Work_Obj</td>
<td>SalesAccountTerritory_Cat</td>
<td>2013/2/11 12:13</td>
<td>2013/2/11 02:45</td>
</tr>
</tbody>
</table>

**Process Details**

<table>
<thead>
<tr>
<th>Work Object Public Unique Identifier</th>
<th>Assignment Status</th>
<th>Message Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>1243213</td>
<td>Successful</td>
<td></td>
</tr>
</tbody>
</table>
An error occurred while loading assignment object Sales_Account_Work_object. There is a mismatch between the view object definition in the assignment configuration and the actual view object definition used during assignment processing. Update and save the assignment object to register it with the latest view object definition.

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1728224</td>
<td>Error</td>
</tr>
<tr>
<td>1982663</td>
<td>Successful</td>
</tr>
<tr>
<td>225030</td>
<td>Successful</td>
</tr>
<tr>
<td>2392053</td>
<td>Successful</td>
</tr>
</tbody>
</table>

The report provides details for implementation to understand if there were errors in the batch assignment being run. This report can also be used to indicate if a particular work object record was processed in a batch assignment process.

## Running Batch Assignment in Diagnostic Mode: Example

You can run your batch assignment in diagnostic mode so that you can view the details of the assignment processing in an output log. This topic provides an example of running lead batch assignment in diagnostic mode.

A profile option called MOW_DIAG_MODE_WO_LIMIT defines the number of work objects that are allowed to be processed in diagnostic mode. The default setting is 1, and you can change the setting before you run batch assignment.

**Note**

For revenue lines which are grouped by opportunity, the MOW_DIAG_MODE_WO_LIMIT value indicates the number of opportunities. For example, the default setting would mean one opportunity which can contain more than one revenue line.

**Scenario**

A sales representative of a company has to follow up on a lead but the lead has not been assigned to his territory. He has requested you, the sales administrator, to investigate the details of territory assignment. You can provide these details by running lead batch assignment in diagnostic mode.

1. Log in to the Oracle Fusion application and select the **Navigator** menu, and then select the **Lead Qualification** menu item.

2. Select **Lead Processing Activities** under the **Tasks** pane.

3. On the Lead Processing Activity page, click the **Create Lead Processing Activity** button.

4. On the Create Lead Processing Activity page:
   a. Select **Assignment** from the **Process Type** list.
b. Enable diagnostic mode by checking the **Diagnostic Mode** check box.

c. Search and select a lead. Note down the lead number value to use in a later step.

d. Select **Immediate** from the **Schedule** list.

e. Click **Submit**.

5. In the Confirmation dialog box, click **OK**.

6. Click the Refresh icon till the process has completed successfully or with an error.

7. Click the Output log icon in the View Log column to view details.

Open the log file in another browser window or tab.

**Note**

The log file format is designed to be viewed in a browser application. If the log file is opened in another application, such as Notepad, the format may not be optimal and the log may be difficult to read.

View the log for details of the assignment processing for the selected lead. You can use the lead number noted down earlier to search in the log file. Review the log for details of the assignment processing.

**Batch Assignment Diagnostic Log: Explained**

When you run batch assignment in diagnostic mode, an output log is generated with details of the assignment processing. You can use these details to troubleshoot any issues with territory assignment. The log helps you understand why certain leads or opportunities were not assigned to your territories as expected.

The table below provides an example of a lead batch assignment diagnostic log along with an explanation of each section of the log.

**Note**

Use the search feature in your log file to search on keywords, such as the error message number, for example 225203, to locate a specific section.

<table>
<thead>
<tr>
<th>Example Log File Entries</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment processing is in diagnostic mode. Assignment results will not be stored and the first 1 work objects will be processed. (MOW-225203)</td>
<td>Provides a summary of the assignment processing and the number of work objects to be processed.</td>
</tr>
</tbody>
</table>

**Note**

You can change the number of work objects allowed to be processed in diagnostic mode via the MOW_DIAG_MODE_WO_LIMIT profile option. The default setting is 1.
Process 123 submitted at 2013/2/19 12:10
started assignment processing of 1 work objects.
(MOW-225172)

<table>
<thead>
<tr>
<th>Work Object = Lead_Work_Object_Lead</th>
</tr>
</thead>
<tbody>
<tr>
<td>Candidate Object = Territory_Candidate_Object_Lead</td>
</tr>
<tr>
<td>Assignment Mode = Territory</td>
</tr>
<tr>
<td>View Criteria = LeadAssignmentDiagnosticVC</td>
</tr>
<tr>
<td>View Criteria Bind Values = BindLeadNumberDiag=LEAD_1527</td>
</tr>
<tr>
<td>Rule Set Group Id = 12221</td>
</tr>
<tr>
<td>Rule Set Group Name =</td>
</tr>
<tr>
<td>Grouping Attribute =</td>
</tr>
<tr>
<td>Replace Team = True?</td>
</tr>
<tr>
<td>Number of Work Objects per Sub Process = 1000</td>
</tr>
<tr>
<td>Maximum Sub Processes per Process = 10</td>
</tr>
<tr>
<td>Metric Logging Interval = 100</td>
</tr>
<tr>
<td>Custom Data =</td>
</tr>
<tr>
<td>Diagnostic Mode = true</td>
</tr>
</tbody>
</table>

Matching request for work object
Lead_Work_Object_Lead with the
identifier Lead_1527 and candidate object
Territory_Candidate_Object_Lead is in process.
(MOW-225169)

| Geography Identifier = 1272833, Sell to Address = 500 Oracle Pkwy, Redwood Shores, CA 94065 USA |
| Party Identifier =12393333, Sales Account Number = 5273 |
| Customer Size Code = MEDIUM |
| Industry = null |
| Named Account Type = N |
| Organization Type = null |
| Inventory Item = 17182823, Product = Green Server |
| Product Group = null |
| AuxiliaryDimension1 = null |
| Partner Identifier = null, Registry Identifier = null |
| Score = 13212 Etc. |

Allows you to confirm the objects being processed in this batch, for example territories being assigned to leads, the type of assignment processing, and the other parameters and their values relevant for this batch process. Indicates the following:

- The process has started.
- Work object being processed and the candidates being found.
- Type of assignment processing:
  - Territory = territory assignment
  - Matching = assignment using rules
  - Score = scoring
  - Classification = ranking or qualification
- View criteria and bind value in leads which determine the set of leads that are included in this batch assignment process.
- Diagnostic mode setting.

Provides a summary of the active assignment attributes and their values that will be used in the processing of this lead.

**Note**

Only a subset of these attributes may be used in the assignment processing of a lead, for example lead ranking rule may only use the score and timeframe attributes.

- Values for the work object.
- Attributes that are null.
- Products available: In the example, there is one product, green server.

Use this information to confirm the data values for the work object that may be used in the assignment processing.
Assignment matching using mapping set Mapping Set 1 is in process. (MOW-225185)

Assignment mapping values were retrieved. (MOW-225211)

Function Code = Geo, GeographyId = 182939393, Sell to Address = 500 Oracle Pkwy, Redwood Shores, CA 94065 USA

Function Code = Acct, PartyId =1232222933333

Function Code = CustSze, CustomerSize =

Function Code = Acct, PartyId =1232222933333

Function Code = Prod, Inventory Item Identifier = 12222, Product = Green Server, Inventory Org Id = 15161717

Partner Identifier = Partner Program Identifier = 17832833

Assignment mapping values were translated to sequence values. (MOW-225212)

Function Code = Geo, Sequence = 1 -1

Function Code = Acct, Sequence = 12223 - 333334, 523712-728299

Function Code = CSize, Sequence = 233 - 27389

Function Code = AcTyp, Sequence = 211 - 211

Function Code = OrgTp, Sequence = 3643 - 3833

Function Code = Indst, Sequence = 233 - 811

Function Code = Prod, Sequence 1221 - 3783

This shows the active assignment mappings that drive territory-based assignment for this lead. The value(s) for each mapping is also shown.

If an unexpected territory or set of territories has previously been assigned to the work object, then confirm that this is the information you expected to be used for this lead.

This section is relevant for Oracle Support to troubleshoot assignment issues.

Candidate matches were identified. Post processing is in progress. (MOW-225210)

 Territory with the Number 282274 was dropped as it is a parent. (MOW-225209)

 Territory with the Number 282312 was dropped due to an exclusion. (MOW-225208)

 Territory with the Number 238424 was dropped due to an exclusion. (MOW-225208)

 Territory with the Number 120238 was dropped as it is a parent. (MOW-225209)

Indicates that matching candidates were found and lists the matching candidates that were dropped since they are either parent candidates or part of excluded territories.

If this lead was assigned previously and the territory you expected was not assigned, then review this list of dropped territories. The territory might have been dropped because it was a parent territory and a lower level territory under this also matched. Alternatively, this territory might have been excluded for this lead (for example, a partner might have rejected this lead) or the territory might have an exclusion coverage that contains the account on this lead.
The final matching candidates for mapping set Sales Account Mapping Set were identified. (MOW-225207)

<table>
<thead>
<tr>
<th>TerritoryNumber</th>
<th>Shows the final list of matching territories.</th>
</tr>
</thead>
<tbody>
<tr>
<td>284844</td>
<td></td>
</tr>
<tr>
<td>284554</td>
<td></td>
</tr>
<tr>
<td>281274</td>
<td></td>
</tr>
<tr>
<td>161238</td>
<td></td>
</tr>
<tr>
<td>210372</td>
<td></td>
</tr>
</tbody>
</table>

Assignment matching using mapping set Mapping Set 2 is in process. (MOW-225185)

<table>
<thead>
<tr>
<th>Shows the progress of assignment matching for a mapping set.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The mapping set Mapping Set 2 was skipped as the conditional attribute PartnerId is blank. (MOW-225206)</td>
</tr>
<tr>
<td>In this example, there are multiple active mapping sets and the next mapping set (for example, Mapping Set 2) has a conditional attribute defined. For this lead, this attribute does not contain a value, and so the territory matching for this mapping set is not needed and therefore not performed.</td>
</tr>
</tbody>
</table>

Assignment matching using mapping set Mapping Set 3 is in process. (MOW-225185)

<table>
<thead>
<tr>
<th>Shows the progress of assignment matching for a mapping set.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The mapping set Mapping Set 3 was skipped as the conditional attribute ProspectAcctId is blank. (MOW-225206)</td>
</tr>
<tr>
<td>In this example, there are multiple active mapping sets and the next mapping set (for example, Mapping Set 3) has a conditional attribute defined. For this lead, this attribute does not contain a value, and so the territory matching for this mapping set is not needed and therefore not performed.</td>
</tr>
<tr>
<td>Assignment of 4 Territory_Candidate_Object_Lead candidates to work object Lead_Work_Object_Lead with the identifier Lead_1527 is in process. (MOW-225167)</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>The existing candidates were identified. (MOW-225200)</td>
</tr>
<tr>
<td>TerritoryNumber = 1628299</td>
</tr>
<tr>
<td>TerritoryNumber = 1019131</td>
</tr>
<tr>
<td>Assigned 4 new Territory_Candidate_Object_Lead candidates to work object Lead_Work_Object_Lead with the identifier Lead_1527. (MOW-225180)</td>
</tr>
<tr>
<td>TerritoryNumber = 327211</td>
</tr>
<tr>
<td>TerritoryNumber = 210372</td>
</tr>
<tr>
<td>TerritoryNumber = 282274</td>
</tr>
<tr>
<td>TerritoryNumber = 284554</td>
</tr>
<tr>
<td>Removed 2 Territory_Candidate_Object_Lead candidates to work object Lead_Work_Object_Lead with the identifier Lead_1527. (MOW-225181)</td>
</tr>
<tr>
<td>TerritoryNumber = 1628299</td>
</tr>
<tr>
<td>TerritoryNumber = 1019131</td>
</tr>
<tr>
<td>Assignment of 4 Territory_Candidate_Object_Lead candidates to work object Lead_Work_Object_Lead is complete. (MOW-225166)</td>
</tr>
<tr>
<td>1 out of 1 work objects were completed. 1 work objects successfully processed. 0 work objects failed. (MOW-225127)</td>
</tr>
</tbody>
</table>

**Purge Batch Assignment Information: Explained**

When a batch assignment job runs, it creates data that helps with the assignment process. Once the job is completed, this data is no longer required and can be purged. Oracle Fusion Assignment Manager enables implementations to purge data. The Fusion Enterprise Scheduling Service (ESS) process **Purge Batch Assignment Information** is used to purge the batch assignment tables based on set parameters. A batch assignment process creates data in two tables MOW_BATCH_ASGN_JOBS and MOW_BATCH_ASGN_JOB_ITEMS. When the size of the batch assignment process is huge, it creates a large number of rows in the MOW_BATCH_ASGN_JOB_ITEMS table. With time, records in these tables grow substantially, especially with large implementations. The purge batch assignment information process helps in clearing old records for successfully completed processes. The process purges data for batch assignment processes for Leads, Opportunities, Sales Accounts and so on.

Implementations may run this process periodically or on an ad-hoc basis. The process has only one parameter **Days to Keep**. The default value is 30.
Specify the number of days worth of assignment batch job and data to keep until the next purge. For example, if you set the parameter to 15 and run the purge batch assignment information process, the process removes all rows related to successfully completed processes in the batch assignment data table that were created before 15 days from the current date.

**Territory Dimension Data Report: Explained**

The territory dimension data report identifies the volume of territory data for each territory dimension and coverage type. You can use information from this report to determine the sequence for each assignment mapping and optimize assignment performance. You must run this report only after you have created and activated your production territories.

Access the Diagnostic Dashboard to generate the territory dimension data report. The report contains one table that shows the number of de-normalized customer account-centric and partner-centric territory records for each territory dimension and coverage type.

The table below shows an example of a territory dimension data report:

<table>
<thead>
<tr>
<th>Function Code</th>
<th>Count for INCLUSION</th>
<th>Count for EXCLUSION</th>
<th>Count for PARTNER_REG</th>
<th>Count for REGULAR</th>
<th>Total Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcTyp</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>305</td>
<td></td>
</tr>
<tr>
<td>Acct</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>313</td>
<td></td>
</tr>
<tr>
<td>Aux1</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td></td>
</tr>
<tr>
<td>Aux2</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td></td>
</tr>
<tr>
<td>Aux3</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td></td>
</tr>
<tr>
<td>CSize</td>
<td>7</td>
<td>3</td>
<td>13</td>
<td>336</td>
<td></td>
</tr>
<tr>
<td>Geo</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Indst</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>268</td>
<td></td>
</tr>
<tr>
<td>OrgTp</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>374</td>
<td></td>
</tr>
<tr>
<td>Prod</td>
<td>2</td>
<td>3</td>
<td>16</td>
<td>171</td>
<td></td>
</tr>
<tr>
<td>Prtnr</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td></td>
</tr>
<tr>
<td>Schnl</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>259</td>
<td></td>
</tr>
</tbody>
</table>

Once the report is generated, you must identify the sequence of mappings as follows:

1. Copy the report over to an excel spreadsheet.
2. Sum up the count for inclusion, exclusion, partner_regular, and regular for each function code.
3. List the function codes in order from lowest total count to highest.

After you list function codes from lowest count to the highest, the report table should look like the following:

<table>
<thead>
<tr>
<th>Function Code</th>
<th>Count for INCLUSION</th>
<th>Count for EXCLUSION</th>
<th>Count for PARTNER_REG</th>
<th>Count for REGULAR</th>
<th>Total Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geo</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Prod</td>
<td>2</td>
<td>3</td>
<td>16</td>
<td>171</td>
<td>192</td>
</tr>
<tr>
<td>Schnl</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>259</td>
<td>285</td>
</tr>
<tr>
<td>Indst</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>268</td>
<td>292</td>
</tr>
<tr>
<td>Acct</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>313</td>
<td>329</td>
</tr>
<tr>
<td>AcTyp</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>305</td>
<td>331</td>
</tr>
<tr>
<td>CSize</td>
<td>7</td>
<td>3</td>
<td>13</td>
<td>336</td>
<td>359</td>
</tr>
<tr>
<td>OrgTp</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>374</td>
<td>398</td>
</tr>
<tr>
<td>Aux1</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td>411</td>
</tr>
<tr>
<td>Aux2</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td>411</td>
</tr>
<tr>
<td>Aux3</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td>411</td>
</tr>
<tr>
<td>Prtnr</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td>411</td>
</tr>
</tbody>
</table>

With this information, you can now update the sequence for each assignment mapping. The function code with the lowest total count, Geo in this example, should be updated to have the sequence 1. The next lowest total count, Prod in this example, should have sequence 2, and so on. You must update the sequence similarly for every mapping in each mapping set, for every work-object and candidate-object combination.
Define Territory Management Configuration

Territory Components: How They Work Together

Territories are used to define the jurisdiction of responsibility of a salesperson. 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.

![Territory Diagram](image)

**Modeling Territories: Things to Consider**

Model your territories to support your sales goals, such as the introduction of a new product, in addition to providing salespeople equitable territories to support their productivity. One salesperson can belong to multiple territories.

**Sales Resource Structure**

You can model territories as hierarchies that are similar to the sales resource hierarchy. For example, the sales manager owns the parent territory and the salespeople who report to the manager own child territories.

**Multiple Sales organizations**

A higher level sales executive owns a parent territory with several child territories owned by senior managers. Each senior manager has a sales organization that focuses on selling to support a particular sales goal.

For example, one senior manager is at the top of a hierarchy of territories that are defined by geography. Another senior manager owns a territory hierarchy and sales organization who sell only to government customers. A third territory hierarchy supports selling the new product line to any customer with interest in the product. The fourth territory hierarchy assigns salespeople to specific important customers. Some of the senior managers create a child territory hierarchy to oversee partners.

**Forecasts and Quotas**

When you model your territories, you also want to think about how you want to manage your forecasts and quotas. Quotas are set and distributed from the top...
of the single overall territory hierarchy down through the levels of territories. A manager owner of the parent territory sets the quotas for the child territories and their owners.

Forecasts roll up the territory hierarchy. What forecasts, pipeline, and closed sales do executives want to monitor? Do they want to see how a new product line is doing? Watch over a country where you just started selling? Compare different industries? You can model your territory hierarchy to support your forecasting and sales analysis needs.

Sales Goals

You can model territories to support specific sales goals.

Products

You can delegate a sales team to selling a particular product. Perhaps you have a group of products that requires specific technical expertise to sell it. Or you want to provide incentives to sell a new product line. Leads and opportunity line items that include a product will be automatically assigned to the territory defined for the product or a parent of the product in the sales catalog.

New Customers

Your sales analysts identified opportunities for finding new customers. You can model territories to support your sales goal to expand your customer base. Perhaps the analysts identified certain industries to be good prospects. You can define territories partly by the industry, and assign a sales team dedicated to pursuing customers in that industry.

You open a new geographic area. Territories not only take care of assigning customers, leads, and opportunities to the salespeople in that region, but also provide the structure for monitoring and analyzing the sales forecasts and results for the new region.

Security and Access

You control access to customers, leads, and opportunities partly with your territory structure.

Salespeople have access to customers, leads, and opportunities that fall within their territories. Salespeople assigned to the parent territory also have access the same access to child territories and on down the hierarchy. To find other topics about security and access, search for customers, leads, opportunities, assignment, and security.

Define Territory Management Dimensions

Territories Defined by Dimensions: Explained

Dimensions define territory boundaries. For example, the geography dimension can be used to define territories by specified countries, cities, and postal codes.
Territory dimensions are used to assign customer accounts, partners, leads, and opportunity line items to the correct territories.

A territory captures every sales account or partner that falls within the defined boundaries of the territory.

Here is a geography example. The sales team assigned to this territory is responsible for all customers in Spain. In this example, the Spain Territory is assigned sales accounts for Customer A, but not Customer B.

**Customers and Partners**

The dimensions used to assign sales accounts or partners to a territory are:

- **Customer**
  
  Use the Included Customers region of the Coverages tab to select individual customers, with or without the customer hierarchy. The selected customer's sales accounts will be assigned to the territory. You can also select customers to be excluded from the territory.
  
  Alternatively, use the Customer dimension in the Dimensional Coverage region to select either customers that are large enough to have a customer hierarchy in Account Center, or customers with at least one sales account with Named Account selected.

- **Geography**

- **Account Type**
  
  Was the sales account designated a named account or not?

- **Customer Size**

- **Industry**

- **Organization Type**

- **Partner**

  Use the Included Partners region of the Coverages tab to select individual partners to be included in the territory. You can also select partners to be excluded from the territory.

Your administrator can also use classification categories to define up to three additional dimensions based on classifications to assign customers or partners.

**Leads and Opportunities**

A territory captures specific leads or deals that are for customers or partners assigned to the territory. Additionally, you can define a territory using information about that deal.
Your territory does not have to be defined using customer information. It can capture all leads and opportunity line items that match dimension definitions. Or you can create a territory with customer boundaries in addition to defining what can be sold to assigned customers.

The additional dimensions used to assign leads and opportunity line items are:

- **Product**
  Groups of products form a hierarchy in the sales catalog.
- **Sales Channel**
  The available sales channels are Direct, Indirect, and Partner.

For example, your company is launching a new product line. One sales manager with a small sales team will specialize in selling only the new line. The sales manager’s territory is defined only by the new product.

One of the salespeople reporting to this manager sells the new product line in Canada. The following figure shows the salesperson’s territory and an opportunity line item assigned to the territory:

**Territory Dimension Administration: Explained**

Dimensions are attributes that define jurisdictional boundaries of territories. For example, the geography dimension can be used to define territories by country or postal code. Every sales account that falls within the defined geography is assigned to the territory and to the sales team for that territory. Territory dimensions are used to assign customer accounts, partners, leads, and opportunity line items to the correct territories. All dimension values combine to define the territory boundaries. For example, if Geography = United States and Customer Size = Large, then the territory will be assigned to only large customers within the United States.

**Dimensions**

The dimensions available to assign sales accounts or partners to a territory are:

- **Customer**
  Use the Included Customers region of the Coverages tab to select individual customers, with or without the customer hierarchy, to be
assigned to the territory. The selected customer’s sales accounts will be assigned to the territory. You can also select customers to be excluded from the territory. Alternatively, use the Customer dimension in the Dimensional Coverage region to select either customers that are large enough to have a customer hierarchy in Account Center, or customers with at least one sales account with Named Account selected.

- Geography
- Account Type
  
  Was the sales account designated a named account or not?
- Customer Size
- Industry
- Organization Type
- Partner

  Use the Included Partners region of the Coverages tab to select individual partners to be included in the territory. You can also select partners to be excluded from the territory.

- Additional auxiliary dimensions based on selected classification categories to match specific customer or partner information.

A territory assigned to a sales account or partner will also be assigned to that customer’s leads and opportunities. The following additional dimensions are available to define boundaries specifically for leads and opportunity line items:

- Product
  
  Groups of products form a hierarchy in the sales catalog.
- Sales Channel

  The available sales channels are Direct, Indirect, and Partner.

You must enable each dimension that the sales organization plans to use for automatically assigning territories to sales accounts, partners, leads, or opportunity line items. The Customer Type, Customer Size, and Sales Channel dimensions are ready to use. Other dimensions require preparation before you enable and use them.

**Dimensions You Can Modify**

The following dimensions require a step to implement them or are ready to use but can be modified:

- Customer Size

  You can change the provided customer sizes within the **Organization Size** lookup code.
- Industry

  The industry hierarchy is from the customer classification module. When you add the Industry dimension in Enable Dimensions and Metrics, you must also select the classification category that you want to use. The available selections include only classification categories belonging to the
Industrial Categories grouping. Your selection will set the profile option Industry Classification Category.

- **Organization Type**
  
  To change the available organization types, edit the ORGANIZATION_TYPE category using Manage Classification Categories.

**Dimensions Requiring Preparation**

See related topics to find out how to prepare the following dimensions:

- Customer
- Geography
- Partner
- Product
- Additional Dimensions

**Preparing the Customer Dimension: Explained**

Customers and their hierarchies, along with related sales accounts, are maintained in the Customer Center application. 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. Only sales accounts are assigned to territories.

You have two methods of using the Customer dimension when defining territories:

1. Select individual customers, with or without customer hierarchies, in the Included Customers region of the Coverages tab. Any customer is eligible for selection. You can also select individual customers to be excluded.

2. Select customers, with or without their customer hierarchies, in the Dimensional Coverage region of the Coverages tab. The only customers eligible for selection are both:
   - Customers part of a customer hierarchy
   - Customers with at least one sales account designated as a Named Account

Perform the following steps to prepare the customer dimension:

1. Use the Account Center or import to add customer and sales account data.

2. Go to the Enable Dimensions and Metrics territory setup task.

3. If you want to enable the Customer dimension to be available in the Dimensional Coverage region of the Coverages tab, then add the Customer dimension in the Dimensions region to enable it.

4. Select Customers in the Inclusions and Exclusions region to enable the Included Customers region of the Coverages tab.
5. Click **Load and Activate** to start the background process. The process loads the customers to become dimension members available for selection when defining territories. The process also loads dimension member data for other enabled dimensions.

6. When changes occur in customer data, you must again run the Load and Activate process to update the Customer territory dimension member data. Or you can perform the maintenance task Run Territory Dimension Synchronization to schedule the process.

7. It is a good practice to run full reassignment processes for customers, leads, and opportunities after updating Customer dimension members in the previous step.

### Preparing the Geography Dimension: Explained

If you plan to assign customers to territories according to the customer’s address, then you need to provide geography data for the Geography dimension members. The source master geography data is also used outside of territory definitions, such as for customer address validation. The same geography hierarchy is used for all applications.

The following additional steps implement existing geography data for use in the Geography dimension:

1. Use the Manage Geographies setup task to enable geography validation, and set geography validation to Error, for every geography level that you intend to use for territory definition for each country. If you have already created addresses before setting up geography validation for a country, then you must execute the Run Maintain Geography Name Referencing task for that country after enabling geography validation to ensure that all your geography elements are validated.

2. You must build one or more territory geography hierarchies using the Manage Territory Geographies setup task. You will use geography elements from the master geography data to form your hierarchy. The hierarchies created in this step become geography dimension members available for defining territories.

   An optional feature of the territory geographies is the use of custom zones. You can create a zone that is a parent of one or more master geographies. For example, you can create an Americas zone that includes several countries. The Americas zone becomes a dimension member that can be selected for defining a territory. Before you create zones, make sure all of your sales organizations agree upon the definition of each zone.

3. Go to the **Enable Dimensions and Metrics** territory setup task.

4. Add Geography to the enabled dimensions.

5. Click **Load and Activate** to start the background process. The process loads the territory geography data to become dimension members available for selection when defining territories. The process also loads dimension member data for other enabled dimensions.

6. When changes occur in geography data, you must again run the Load and Activate process to update the Geography territory dimension member data.
data. Or you can perform the maintenance task Run Territory Dimension Synchronization to schedule the process.

7. It is a good practice to run full reassignment processes for customers, leads, and opportunities after updating Geography dimension members in the previous step.

**Preparing the Partner Dimension: Explained**

A partner is an organization party with a partner profile associated and an assigned Partner usage. Partners are defined in the Account Center.

Similar to direct sales, channel managers have corresponding sales territories. The territories are defined by partner-centric coverages. The partner territory coverages assign partners in the following ways:

1. Defining the characteristics of the end customer, irrespective of which partner is associated with the transaction
2. Defining characteristics of the partner, such as where the partner is located or the type of the partner (reseller, system integrator, distributor)
3. Selecting specific partners to be included in or excluded from the territory

The first two definitions require no preparation. You must enable inclusions and exclusions of partners using the Enable Dimensions and Metrics setup task to use the third territory coverage method.

**Preparing the Product Dimension: Explained**

The Product dimension captures leads and opportunity line items. You create a hierarchy of products and product groups in the sales catalog. Salespeople can then select the products or product groups when creating leads or opportunities. When you include a product group in a territory coverage, the territory then captures leads and opportunity line items containing a product within the hierarchy of the product group included in the territory coverage.

You must create the sales catalog before enabling the Product dimension. It is a good practice to initially set up your sales catalog using product groups, and then add individual products or inventory items if needed.

Perform the following steps to prepare the Product dimension:

1. Go to the Manage Product Groups setup task.
2. Create your root product group with the following settings:
   - Active
   - Root Catalog
   - Locked
   - Allow Duplicate Children deselected
3. Use the Subgroups tab to create product groups within the root catalog. You can create a hierarchy of product groups.

4. Publish your sales catalog.

5. Use the Manage Product Group Usage setup task to add your root product group as the Base usage.

6. Go to the Enable Dimensions and Metrics territory setup task.

7. Add Product to the enabled dimensions.

8. Click Load and Activate to start the background process. The process loads the sales catalog data to become dimension members available for selection when defining territories. The process also loads dimension member data for other enabled dimensions.

9. When changes occur in the sales catalog, you must again run the Load and Activate process to update the Product territory dimension member data. Or you can perform the maintenance task Run Territory Dimension Synchronization to schedule the process.

10. It is a good practice to run full reassignment processes for leads and opportunities after updating Product dimension members in the previous step.

Creating Additional Dimensions: Explained

You can create up to three additional dimensions that will match with customer attributes.

Auxiliary Dimension 1, 2, or 3

You can define up to three customer auxiliary dimensions based on the customer classification model. Define classification categories first and associate them to the Customer Categories grouping.

The following steps must be done before you can enable auxiliary dimensions:

1. Use the Manage Classification Categories task to create a new classification category. You can allow parent code assignment. If you allow multiple class code assignments, then the classification that is designated as Primary in the customer record is the one that is matched to assign the customer to a territory.

2. Add classification codes for the new category. You can form a hierarchy with the codes or not.

3. Use the Manage Classification Groups task and search for the CUSTOMER_GROUP category group code. Edit the group and add your new classification category to the group.

4. Add an auxiliary customer dimension in Enable Dimensions and Metrics and select the correct classification category that you created. Then this classification category becomes the source for dimension members for the auxiliary dimension.

5. Load and activate your newly enabled dimensions.
Populating Dimension Members: Explained

Dimension members are populated and synchronized in Oracle Business Intelligence Suite Enterprise Edition Plus either through the data warehouse, or without the use of data warehouse. You must repopulate dimension members when source data changes, such as changes in the product catalog and the addition of new customers.

Cube

Territory Management generates the cube based on the enabled dimensions and their members. In the case of the data warehouse option, Territory Management loads the cube with dimension members and metrics directly from the data warehouse. In the case of the non-data warehouse option, the application loads the cube from Fusion Applications. The cube provides metrics information for defined territories.

Sequence

Following is the sequence for populating dimensions for use in defining territories:

1. The accounting calendar should be set up and the profile option CRM Common Calendar set. This calendar is used throughout CRM.
2. Set up source data for the dimensions you plan to use. This includes:
   - Master geography and territory geography zone hierarchies
   - Customers with a sell-to address
   - Sales catalog
   - Customer sizes lookup
   - Classification codes for the Organization Type classification category
   - Industry classification category and codes
   - Customer classification categories and codes for auxiliary dimensions
3. If you are using data warehouse, then run ETL to populate the data warehouse with the dimension members and transactional data.
4. Enable the dimensions and metrics that you plan to use. Load and activate the dimension members.

Non-Data Warehouse Option

When you do not use data warehouse, such as in Oracle Sales Cloud Service, 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

(On-Premise implementations only.) 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.

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.

**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.

**Customer**
You have a few key customers that should belong to top salespeople. Use the customer dimension to create territories for individual customers, including their customer hierarchies.

**Account Type**
You want to assign major sales accounts to Named accounts territories. A named account territory can have child territories identified by additional 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.

**Territory Dimension Data Report: Explained**

The territory dimension data report identifies the volume of territory data for each territory dimension and coverage type. You can use information from this
report to determine the sequence for each assignment mapping and optimize assignment performance. You must run this report only after you have created and activated your production territories.

Access the Diagnostic Dashboard to generate the territory dimension data report. The report contains one table that shows the number of de-normalized customer account-centric and partner-centric territory records for each territory dimension and coverage type.

The table below shows an example of a territory dimension data report:

<table>
<thead>
<tr>
<th>Function Code</th>
<th>Count for INCLUSION</th>
<th>Count for EXCLUSION</th>
<th>Count for PARTNER_REGULAR</th>
<th>Count for REGULAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcTyp</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>305</td>
</tr>
<tr>
<td>Acct</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>313</td>
</tr>
<tr>
<td>Aux1</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
</tr>
<tr>
<td>Aux2</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
</tr>
<tr>
<td>Aux3</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
</tr>
<tr>
<td>CSize</td>
<td>7</td>
<td>3</td>
<td>13</td>
<td>336</td>
</tr>
<tr>
<td>Geo</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Indst</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>268</td>
</tr>
<tr>
<td>OrgTp</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>374</td>
</tr>
<tr>
<td>Prod</td>
<td>2</td>
<td>3</td>
<td>16</td>
<td>171</td>
</tr>
<tr>
<td>Prtnr</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
</tr>
<tr>
<td>Schnl</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>259</td>
</tr>
</tbody>
</table>

Once the report is generated, you must identify the sequence of mappings as follows:

1. Copy the report over to an excel spreadsheet.
2. Sum up the count for inclusion, exclusion, partner_regular, and regular for each function code.
3. List the function codes in order from lowest total count to highest.

After you list function codes from lowest count to the highest, the report table should look like the following:

<table>
<thead>
<tr>
<th>Function Code</th>
<th>Count for INCLUSION</th>
<th>Count for EXCLUSION</th>
<th>Count for PARTNER_REG</th>
<th>Count for REGULAR</th>
<th>Total Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geo</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>Prod</td>
<td>2</td>
<td>3</td>
<td>16</td>
<td>171</td>
<td>192</td>
</tr>
<tr>
<td>Schnl</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>259</td>
<td>285</td>
</tr>
<tr>
<td>Indst</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>268</td>
<td>292</td>
</tr>
<tr>
<td>Acct</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>313</td>
<td>329</td>
</tr>
<tr>
<td>AcTyp</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>305</td>
<td>331</td>
</tr>
<tr>
<td>CSize</td>
<td>7</td>
<td>3</td>
<td>13</td>
<td>336</td>
<td>359</td>
</tr>
<tr>
<td>OrgTp</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td>374</td>
<td>398</td>
</tr>
<tr>
<td>Aux1</td>
<td>7</td>
<td>3</td>
<td>16</td>
<td>385</td>
<td>411</td>
</tr>
</tbody>
</table>
With this information, you can now update the sequence for each assignment mapping. The function code with the lowest total count, Geo in this example, should be updated to have the sequence 1. The next lowest total count, Prod in this example, should have sequence 2, and so on. You must update the sequence similarly for every mapping in each mapping set, for every work-object and candidate-object combination.

**Territory Export and Import**

**Territory Export and Import: Overview**

You can export any branch of territories to a spreadsheet, perform edits, and import your changes into a territory proposal. You can also export all territories from your test environment and import them to your active environment, replacing all records. If you use the provided spreadsheet format, then you can add new territories by importing the spreadsheets.

Four territory objects are exported and imported using the following four CSV files:

- **Header**
  The Header file is always required for an import. The file includes the territory name, parent territory, and territory owner.

- **Resources**
  The resources file is optional. It includes the territory name, resource name, resource function, and resource e-mail address. It is used to import additional territory team members, besides the owner.

- **Lines of Business**
  The lines of business file is optional. It contains the territory name and the line of business for the territory.

- **Coverages**
  The coverages file is optional. It contains the territory name and all coverage definitions for each territory. All dimension members needed to define the territory are listed. For every enabled dimension that is not included in the file, the dimension is given the value Any. Included and excluded customers or partners are also listed.

Export and import are available from the Actions list for the territories table in the Active Territories screen. The Territory Proposal screen has only export actions available.

---

**Note**
You will see two selections for both export and import: one for editing territories and one for moving territories to and from another environment.

**Territory Coverage Import File Reference**

You can export and import territories using a spreadsheet, from the Actions menu in the territories table in the Active Territories screen. If you use the provided spreadsheet format, then you can add new territories by importing the spreadsheets. This topic describes the spreadsheet and corresponding table columns for the TERR_COVERAGE.csv file. This file is optional for imports.

Import and export use a zip file that contains the following four CSV files:

- Territory Header file (TERR_HEADER.csv)
- Territory Resource file (TERR_RESOURCE.csv)
- Line of Business file (TERR_LOB.csv)
- Territory Coverage file (TERR_COVERAGE.csv)

**TERR_COVERAGE.csv File**

The following table lists the columns included in the Territory Coverage file along with descriptions and whether or not the column is used during the import process:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Import?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display_Original_Row_Number</td>
<td>This column is populated only in the rejected data file in case of import errors. It corresponds to the original line number in the input file used by the import process.</td>
<td>No</td>
</tr>
<tr>
<td>Language_Code</td>
<td>The language code used for translatable values. In the export file, this is based on the language of the user who triggered the export. Objects, names, and keys that are translatable will be extracted from records in translation tables.</td>
<td>Yes</td>
</tr>
<tr>
<td>Territory_Number</td>
<td>Territory number as in the source environment. For new territories it can be specified or can be left blank. If blank, then the system will generate a number at the time of import.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Territory_Name</td>
<td>Territory name. Used only for a reference.</td>
<td>No</td>
</tr>
<tr>
<td>Coverage_Type</td>
<td>The coverage type such as: Dimensional, Inheritance, or Override. Valid names are in the lookup type MOT_TERR_COVERAGE_TYPE.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Define Territory Management Configuration

#### Valid values are:
- Customer
- Customer Auxiliary 1
- Customer Auxiliary 2
- Customer Auxiliary 3
- Customer Size
- Geography
- Industry
- Account Type
- Organization Type
- Partner
- Product
- Sales Channel

#### Dimension Member Rules

Use the rules in the following table when entering dimension member keys or names:

<table>
<thead>
<tr>
<th>Dimension Name</th>
<th>Comment</th>
<th>Dimension_Member_Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Only one of two values are allowed. Dimension_Member_Name does not need to be specified.</td>
<td>Named / Not Named</td>
</tr>
<tr>
<td>Account Auxiliary 1..3</td>
<td>Enter a value as seen in territory UIs. Dimension_Member_Name does not need to be specified.</td>
<td>Value as seen in UIs.</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
<td>Value</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Customer</td>
<td>Enter OS+OSR Number (if known) or Registry ID (as seen in TM Uis). OS stands for Original System. OSR stands for Original System Reference. OS+OSR exists only if customer data was imported from an external system. Dimension_Member_Name does not need to be specified.</td>
<td>OS+OSR Number or Registry ID (as seen in UIs)</td>
</tr>
<tr>
<td>Customer Size</td>
<td>Enter a value as seen in territory Uls. Dimension_Member_Name does not need to be specified.</td>
<td>Value as seen in UIs.</td>
</tr>
<tr>
<td>Geography</td>
<td>If you know that a specific geography name is unique, then enter a value as seen in territory Uls, for example, Boston. If unsure, or if you know that the name is not unique, enter the full path such as United States<del>New York</del>New York. The full path is made up of geography members as seen in UIs concatenated with ~ character. Dimension_Member_Name does not need to be specified.</td>
<td>Value as seen in territory Uls, or a full path if the geography name is not unique.</td>
</tr>
<tr>
<td>Industry</td>
<td>Enter a value as seen in territory UIs. Dimension_Member_Name does not need to be specified.</td>
<td>Value as seen in UIs.</td>
</tr>
<tr>
<td>Organization Type</td>
<td>Enter a value as seen in territory UIs. Dimension_Member_Name does not need to be specified.</td>
<td>Value as seen in UIs.</td>
</tr>
<tr>
<td>Partner</td>
<td>Enter OS+OSR Number (if known) or Registry ID (as seen in territory UIs). OS stands for Original System. OSR stands for Original System Reference. OS+OSR exists only if partner data was imported from an external system. Dimension_Member_Name does not need to be specified.</td>
<td>OS+OSR Number or Registry ID (as seen in UIs)</td>
</tr>
<tr>
<td>Column Name</td>
<td>Description</td>
<td>Import?</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Display_Original_Row_Number</td>
<td>This column is populated only in the rejected data file in case of import errors. It corresponds to the original line number in the input file used by the import process.</td>
<td>No</td>
</tr>
</tbody>
</table>

**Product**

Enter the product name (in the Dimension_Member_Name column) as seen in TM UIs. Leave Dimension_Member_Key column empty. If this does not result in unique identification of the product dimension member, then the following will need to be done: For product group: Enter Product Group Internal Name (exposed in the Order Capture application) into the Dimension_Member_Key column. For product item: Enter Item Number (exposed in the Product Model application) into the Dimension_Member_Key column. When Dimension_Member_Key column is specified, the value in Dimension_Member_Name column is ignored.

**Sales Chanel**

Enter a value as seen in territory UIs. Dimension_Member_Name does not need to be specified.

**Territory Line of Business Import File Reference**

You can export and import territories using a spreadsheet, from the Actions menu in the territories table in the Active Territories screen. If you use the provided spreadsheet format, then you can add new territories by importing the spreadsheets. This topic describes the spreadsheet and corresponding table columns for the TERR_LOB.csv file. This file is optional for imports.

Import and export use a zip file that contains the following four CSV files:
- Territory Header file (TERR_HEADER.csv)
- Territory Resource file (TERR_RESOURCE.csv)
- Line of Business file (TERR_LOB.csv)
- Territory Coverage file (TERR_COVERAGE.csv)

**TERR_LOB.csv File**

The following table lists the columns included in the Territory Line of Business file along with descriptions and whether or not the column is used during the import process:
<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Import?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language_Code</td>
<td>The language code used for translatable values. In the export file, this is based on the language of the user who triggered the export. Objects, names, and keys that are translatable will be extracted from records in translation tables.</td>
<td>Yes</td>
</tr>
<tr>
<td>Territory_Number</td>
<td>Territory number as in the source environment. For new territories it can be specified or can be left blank. If blank, then the system will generate a number at the time of import.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Territory_Name</td>
<td>Territory name. Used only for a reference.</td>
<td>No</td>
</tr>
<tr>
<td>Line_of_Business</td>
<td>The name of the line of business. Valid names are in the lookup type MOT_LINE_OF_BUSINESS.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

## Territory Resource Import File Reference

You can export and import territories using a spreadsheet, from the Actions menu in the territories table in the Active Territories screen. If you use the provided spreadsheet format, then you can add new territories by importing the spreadsheets. This topic describes the spreadsheet and corresponding table columns for the TERR_RESOURCE.csv file. This file is optional for imports.

Import and export use a zip file that contains the following four CSV files:

- Territory Header file (TERR_HEADER.csv)
- Territory Resource file (TERR_RESOURCE.csv)
- Line of Business file (TERR_LOB.csv)
- Territory Coverage file (TERR_COVERAGE.csv)

### TERR_RESOURCE.csv File

The following table lists the columns included in the Territory Resource file along with descriptions and whether or not the column is used during the import process:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Import?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display_Original_Row_Number</td>
<td>This column is populated only in the rejected data file in case of import errors. It corresponds to the original line number in the input file used by the import process.</td>
<td>No</td>
</tr>
<tr>
<td>Column Name</td>
<td>Description</td>
<td>Import Requirement</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Language_Code</td>
<td>The language code used for translatable values. In the export file, this is based on the language of the user who triggered the export. Objects, names, and keys that are translatable will be extracted from records in translation tables.</td>
<td>Yes</td>
</tr>
<tr>
<td>Territory_Number</td>
<td>Territory number as in the source environment. For new territories it can be specified or can be left blank. If blank, then the system will generate the number at the time of import.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Territory_Name</td>
<td>Territory name. Used only for a reference.</td>
<td>No</td>
</tr>
<tr>
<td>Resource_Email</td>
<td>Resource e-mail address. It is also the resource’s identifier.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Resource_Name</td>
<td>Resource name. Used as a reference only.</td>
<td>No</td>
</tr>
<tr>
<td>Resource_Function</td>
<td>The name of the role (function) that the resource plays on the territory team, such as Salesperson or Legal. Valid names are in the lookup type MOT_TEAM_MEMBER_FUNCTION.</td>
<td>Yes</td>
</tr>
<tr>
<td>Resource_Administrator_Y_N</td>
<td>Is the resource a territory administrator? (Y for yes, N for no.)</td>
<td>Yes</td>
</tr>
<tr>
<td>Resource_Fcast_Delegate_Y_N</td>
<td>Is the resource a forecasting delegate? (Y for yes, N for no.)</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Territory Header Import File Reference**

You can export and import territories using a spreadsheet, from the Actions menu in the territories table in the Active Territories screen. If you use the provided spreadsheet format, then you can add new territories by importing the spreadsheets. This topic describes the spreadsheet and corresponding table columns for the TERR_HEADER.csv file. This file is required for imports. Import and export use a zip file that contains the following four CSV files:

- Territory Header file (TERR_HEADER.csv)
- Territory Resource file (TERR_RESOURCE.csv)
- Line of Business file (TERR_LOB.csv)
- Territory Coverage file (TERR_COVERAGE.csv)

**TERR_HEADER.csv File**

The following table lists the columns included in the Territory Header file along with descriptions and whether or not the column is used during the import process:
<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Import?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal_Use</td>
<td>Used only in the case of test to production export and import.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Original_Row_Number</td>
<td>This column is populated only in the rejected data file in case of import errors. It corresponds to the original line number in the input file used by the import process.</td>
<td>No</td>
</tr>
<tr>
<td>Language_Code</td>
<td>The language code used for translatable values. In the export file, this is based on the language of the user who triggered the export. Objects, names, and keys that are translatable will be extracted from records in translation tables.</td>
<td>Yes</td>
</tr>
<tr>
<td>Proposal_Number</td>
<td>The proposal number to which a given territory should be added. If specified, and the proposal already exists and is not activated, then the territory will be added to the proposal. If specified, and the proposal does not exist yet, then it will be created (with the name and number as in this file). If left blank, then the territory will be added to a new, system generated proposal.</td>
<td>Yes</td>
</tr>
<tr>
<td>Proposal_Name</td>
<td>The proposal name to which a given territory should be added. It is used only when a proposal is created (there are no proposals with the number, as per previous entry, in the system). This name is ignored if the proposal (with the number) already exists, or if the proposal number was not specified.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Added_To_Proposal_Y_N</td>
<td>Indicates if the territory was added to the proposal at the time of export. The value is Y if, in the case of exporting from a proposal, the territory is added. In all other cases the value is N. This is for information purposes only. This field is not imported.</td>
<td>No</td>
</tr>
<tr>
<td>Action</td>
<td>Used only for the export-edit-import flow. The valid values are DELETE or REPLACE.</td>
<td>Yes</td>
</tr>
<tr>
<td>Territory_Number</td>
<td>Territory number as in the source environment. For new territories it can be specified or can be left blank. If blank, then the system will generate a number at the time of import.</td>
<td>Yes</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Territory_Name</td>
<td>Territory name.</td>
<td>Yes</td>
</tr>
<tr>
<td>Parent_Territory_Number</td>
<td>The number of the parent territory. This is used to define territory hierarchy.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Parent_Territory_Name</td>
<td>The name of the parent territory. This is for information purposes only. This field is not imported.</td>
<td>No</td>
</tr>
<tr>
<td>Territory_Type</td>
<td>The name of the territory type, such as Prime or Overlay. Valid names are in the lookup type MOT_TERRITORY_TYPE.</td>
<td>Yes</td>
</tr>
<tr>
<td>Source_Territory_Number</td>
<td>The number of the source (inherited) territory. Used to define territory inheritance.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Source_Territory_Name</td>
<td>The name of the source (inherited) territory. Used to define territory inheritance. This is for information purposes only. This field is not imported.</td>
<td>No</td>
</tr>
<tr>
<td>Partner_Program_Name</td>
<td>Name of the partner program.</td>
<td>Yes</td>
</tr>
<tr>
<td>Territory_Partner_Number</td>
<td>Unique identifier (Registry ID) of the territory partner.</td>
<td>Yes</td>
</tr>
<tr>
<td>Display_Territory_Partner_Name</td>
<td>Name of the territory partner. This is for information purposes only. This field is not imported.</td>
<td>No</td>
</tr>
<tr>
<td>Coverage_Model</td>
<td>The name of the coverage model, such as Sales Account Centric or Partner Centric. Valid names are in the lookup type MOT_TERR_COVERAGE_MODEL.</td>
<td>Yes</td>
</tr>
<tr>
<td>Description</td>
<td>Territory description.</td>
<td>Yes</td>
</tr>
<tr>
<td>Fcast_Participation</td>
<td>The name of the forecast participation, such as Revenue or Nonrevenue. Valid names are in the lookup type MOT_TERR_FCST_PARTICIPATION.</td>
<td>Yes</td>
</tr>
<tr>
<td>Fcast_by_Parent_Territory_Y_N</td>
<td>Forecasted by parent (Y for yes, N for no).</td>
<td>Yes</td>
</tr>
<tr>
<td>Eligible_for_Quota_Y_N</td>
<td>Eligible for Quota (Y for yes, N for no).</td>
<td>Yes</td>
</tr>
<tr>
<td>Revision_Reason</td>
<td>The name of the reason for the quota revision, such as New Territory, Owner Changed. Valid names are in the lookup type MOT_QUOTA_REASON.</td>
<td>Yes</td>
</tr>
<tr>
<td>Revision_Description</td>
<td>Quota revision description.</td>
<td>Yes</td>
</tr>
<tr>
<td>Revised_Quota_Y_N</td>
<td>Indicates territory quota needs to be revised, due to a change in the territory (Y for yes, N for no).</td>
<td>Yes</td>
</tr>
<tr>
<td>Owner_Email</td>
<td>Owner e-mail address. This is also the owner's identifier.</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Display_Owner_Name</strong></td>
<td><strong>Owner name.</strong></td>
<td>No</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------</td>
<td>----</td>
</tr>
<tr>
<td><strong>Owner_Function</strong></td>
<td>The name of the role (function) that the owner plays on the territory team, such as Salesperson or Legal. Valid names are in the lookup type MOTTEAM_MEMBER_FUNCTION</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Owner_Administrator_Y_N</strong></td>
<td>Is the owner a territory administrator? (Y for yes, N for no.)</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Owner_Forecasting_Delegate_Y_N</strong></td>
<td>Is the owner a forecasting delegate? (Y for yes, N for no.)</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Define Lead Management

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

**Implementing Lead Management: Critical Choices**

This topic lists some of the considerations to take into account when planning your implementation of Oracle Fusion Lead Management. No setup tasks are mandatory for Lead Management. You can manage the necessary configuration, integration, and metadata settings through profile options and lookups to tailor lead management functionality to match your business requirements.

**Job Roles and Associated Duty Roles**

The Sales Lead Processing Duty gives access to all the underlying processing duties and processing tasks. The Sales Lead Qualification duty provides access to all underlying qualification duties and qualification tasks.

**Profile Option Decision Points**

Profile options are configurable options that affect application operations. The following table provides you with some decisions that you might need to make and the associated profile option that you access to make the appropriate change. If you want to change the default setup, decisions include:

<table>
<thead>
<tr>
<th>Decision Question</th>
<th>Profile Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which assignment rules do you want to use to assign individual resources to the lead sales team?</td>
<td>Manage Sales Lead Assignment Rules</td>
</tr>
<tr>
<td>Which assignment rules do you want to use to assign a territory team to a lead?</td>
<td>Assignment Rule for Territory-Based Lead Assignment</td>
</tr>
<tr>
<td></td>
<td>Rule for Rule-Based Lead Assignment</td>
</tr>
<tr>
<td>Which lead ranking rules do you want to use?</td>
<td>Assignment Rule for Ranking Leads</td>
</tr>
<tr>
<td>Question</td>
<td>Answer</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Which lead qualification rules do you want to use?</td>
<td>Assignment Rule for Qualifying Leads</td>
</tr>
<tr>
<td>Which leads scoring rules do you want to use?</td>
<td>Assignment Rule for Scoring Leads</td>
</tr>
<tr>
<td>Do you want to display the assessment tab in the Edit Lead page?</td>
<td>Lead Assessment Enabled</td>
</tr>
<tr>
<td>Which assessment template do you want to use for lead qualification?</td>
<td>Lead Assessment Template Default</td>
</tr>
<tr>
<td></td>
<td>Advanced Lead Qualification Enabled</td>
</tr>
<tr>
<td></td>
<td>Lead Qualification Template</td>
</tr>
<tr>
<td>How many days should you look back when querying leads in Search?</td>
<td>Lead Query Maximum Number of Days</td>
</tr>
<tr>
<td></td>
<td>Lead Query Default Number of Days</td>
</tr>
<tr>
<td></td>
<td>Lead Query Warning Threshold Number of Days</td>
</tr>
</tbody>
</table>

**Lookup Decision Points**

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. You might want to change or add some of the values that are used in the lookup tables. Some of the values you can change include:

<table>
<thead>
<tr>
<th>Lookup</th>
<th>Description and Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead Retire Reason</td>
<td>Possible reasons for retiring leads. Retired leads are considered closed leads. Values include:</td>
</tr>
<tr>
<td></td>
<td>Duplicate lead or No purchase interest Lead</td>
</tr>
<tr>
<td>Reject Reason</td>
<td>Possible reasons specified for rejecting leads. Rejected leads can be reassigned or retired. Values include:</td>
</tr>
<tr>
<td></td>
<td>Duplicate lead, Failed to reach contact, and Incorrect data</td>
</tr>
<tr>
<td>Lead Time Frame</td>
<td>Lead cycle duration that usually coincides with a typical sales cycle duration for products and services offered. Values include:</td>
</tr>
<tr>
<td></td>
<td>3 months, 6 months, 9 months, 12 months, 15 months, and 18 months</td>
</tr>
<tr>
<td>Lead Reassignment Reason</td>
<td>Possible reasons specified for reassigning leads. Values include:</td>
</tr>
<tr>
<td></td>
<td>No activity, Other, and Workload</td>
</tr>
<tr>
<td>Lead Rank</td>
<td>Lead rank values used as a measure of lead quality and prioritization. Values include:</td>
</tr>
<tr>
<td></td>
<td>Cool, Hot, and Warm</td>
</tr>
<tr>
<td>Lead Channel</td>
<td>Source channel responsible for lead generation. Values include:</td>
</tr>
<tr>
<td></td>
<td>Direct mail, E-Mail, Fax, Phone, Sales visit, Web, Wireless message, Prediction, Model based prediction, and Rule based prediction</td>
</tr>
</tbody>
</table>
Configure Assignment Manager for Lead Processing

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:

<table>
<thead>
<tr>
<th>Lead Work Object Name</th>
<th>Represents ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales Lead</td>
<td>A sales lead.</td>
</tr>
<tr>
<td>Sales Lead Partner</td>
<td>A partner associated with the lead or associated with the marketing campaign that generated the lead.</td>
</tr>
<tr>
<td>Sales Lead Partner Type</td>
<td>The primary sales channel associated with the lead or associated with the marketing campaign that generated the lead.</td>
</tr>
</tbody>
</table>

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>
</tbody>
</table>
Attribute

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.

**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>• SorceCode</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>
</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.

Configure Sales Lead Assignment Rules

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.

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 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.

**Lead Qualification Template Profile**

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 normalized 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.

**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.

Resulting Task Template
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. 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>• Full</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• View only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></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>Table Title</td>
<td>Details</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------------------</td>
<td>-------------------------------------------------------------------------</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>• Direct mail</td>
<td>Source channel responsible for lead generation.</td>
</tr>
<tr>
<td></td>
<td>• E-Mail</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Fax</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Phone</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Sales visit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Web</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Wireless message</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Prediction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Model based prediction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rule based prediction</td>
<td></td>
</tr>
<tr>
<td>Lead Rank</td>
<td>• Cool</td>
<td>Lead rank values used as a measure of lead quality and prioritization.</td>
</tr>
<tr>
<td></td>
<td>• Hot</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Warm</td>
<td></td>
</tr>
<tr>
<td>Lead Reassignment Reason</td>
<td>• No activity</td>
<td>Possible reasons specified for reassigning leads.</td>
</tr>
<tr>
<td></td>
<td>• Other</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Workload</td>
<td></td>
</tr>
<tr>
<td>Lead Registered Status</td>
<td>• No</td>
<td>Status that indicates whether or not a partner lead is registered.</td>
</tr>
<tr>
<td></td>
<td>• Yes</td>
<td></td>
</tr>
<tr>
<td>Lead Processing Activity Schedule Mode</td>
<td>• Immediate</td>
<td>Scheduling options for lead processing activities.</td>
</tr>
<tr>
<td></td>
<td>• Schedule date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Repeats</td>
<td></td>
</tr>
<tr>
<td>Lead Processing Activity Progress Status</td>
<td>• Completed</td>
<td>Progress details indicating lead current and end processing status.</td>
</tr>
<tr>
<td></td>
<td>• Completed with error</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• New</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• In progress</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Scheduled</td>
<td></td>
</tr>
<tr>
<td>Lead Status</td>
<td>• Converted</td>
<td>Lead status values based on specific actions performed on a lead. Used to mark the milestones in the lead life cycle.</td>
</tr>
<tr>
<td></td>
<td>• Qualified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Retired</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Unqualified</td>
<td></td>
</tr>
<tr>
<td>Lead Time Frame</td>
<td>• 3 months</td>
<td>Lead cycle duration that usually coincides with a typical sales cycle duration for products and services offered.</td>
</tr>
<tr>
<td></td>
<td>• 6 months</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 9 months</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 12 months</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 15 months</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 18 months</td>
<td></td>
</tr>
<tr>
<td>Lead Reject Reason</td>
<td>• Duplicate lead</td>
<td>Possible reasons specified for rejecting leads. Rejected leads can be reassigned or retired.</td>
</tr>
<tr>
<td></td>
<td>• Failed to reach contact</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Incorrect data</td>
<td></td>
</tr>
<tr>
<td>Lead Retire Reason</td>
<td>• Duplicate lead</td>
<td>Possible reasons for retiring leads. Retired leads are considered closed leads.</td>
</tr>
<tr>
<td></td>
<td>• No purchase interest</td>
<td></td>
</tr>
<tr>
<td>Lead Process Scheduling Operators</td>
<td>• Equal to</td>
<td>Operators used for lead processing scheduling options.</td>
</tr>
<tr>
<td></td>
<td>• Greater than</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Greater than or equal to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Less than</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Less than or equal to</td>
<td></td>
</tr>
<tr>
<td>Recommended Lead Actions Type</td>
<td>• Accept leads</td>
<td>Type of actions to perform on recommended leads.</td>
</tr>
<tr>
<td></td>
<td>• Create personal campaign</td>
<td></td>
</tr>
<tr>
<td>Lead Processing Repeat Frequency</td>
<td>• Days</td>
<td>The time intervals between lead processing activities.</td>
</tr>
<tr>
<td></td>
<td>• Months</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Weeks</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Years</td>
<td></td>
</tr>
</tbody>
</table>
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
- Lead Searches

**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>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>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>
</tbody>
</table>
### Lead Searches

The following table lists the profile options used to manage sales lead searches.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Top Sales Account Attribute Minimum for Sales Targeting</strong></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><strong>Top Sales Accounts Attribute for Sales Targeting</strong></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><strong>Top Sales Accounts for Sales Targeting</strong></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>

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Lead Query Maximum Number of Days                | Specify the maximum number of days allowed when searching leads based on the creation date range.  
The value for the maximum number of days must be a positive number. If the value is not a positive number or left blank, the value of 360 days is used. |
| Lead Query Default Number of Days                | Specify the default number of days used to derive the creation date range for searching leads. The search criteria is derived by using the current date for the creation end date value and the profile option to derive the creation start date value.  
For example, if you want leads for the last 15 days to be the default query behavior, then set the profile value to 15. The default creation date search criteria will have an end date for the current date and the start date will be 15 days prior to the current date. The values can be overridden in the search criteria.  
The value for the default number of days must be a positive number. If the value is not a positive number or left blank, the value of 30 days is used.  
This profile option can be defined at both the site and user levels. |
| Lead Query Warning Threshold Number of Days      | Specify the number of days for the lead creation date search criteria before issuing an alert to the user.  
To disable the warning, delete the profile option value. |
| Require Additional Criteria for Lead Search Enabled | Enable additional search criteria, other than creation date, when searching across all leads, or for searches relying on hierarchy rollups. |
Define Common Marketing Configuration

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.
Importing Marketing Standard URLs: Explained

You can create or update marketing standard URLs by navigating to Setup and Maintenance and selecting the Manage Marketing Standard URLs task or by importing data through interface tables. To import marketing standard URLs, you use the tool of your preference to load the data then use scheduled processes to import the data. Having a good understanding of the application object, interface table, and destination table will help you prepare your import data.

Consider the following when importing marketing standard URLs:

- Import interface table, destination table, and application object
- Scheduled processes

Import Interface Table, Destination Table, and Application Object

Importing marketing standard URLs involves one interface table and one destination table.

**Note**

A spreadsheet containing detailed information about the interface table, including the interface table columns, defaults, and validations, is available from the Oracle Enterprise Repository by searching on the interface table name or initiating a search using the FusionApps: Interface Table asset type.

The following table lists the interface table, destination table, and resulting application object:

<table>
<thead>
<tr>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKT_IMP_URLS</td>
<td>MKT_TM_URLS</td>
<td>Marketing Standard URLs</td>
</tr>
</tbody>
</table>

To obtain the unique IDs of existing application data, use the Define Data Export Setup and Maintenance task list.

Scheduled Processes

Navigate to Scheduled Processes to schedule the import of data from the interface tables to the destination tables.

The following table displays the process you can schedule to import marketing standard URLs:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Process Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BulkURLsJob</td>
<td>Import Marketing Standard URLs</td>
</tr>
</tbody>
</table>

Deep Links: Explained

Oracle Fusion CRM Applications enables deep linking to specific CRM details pages from external sources, such as from customer e-mails or customer portal.
pages. In general, a deep link is a hyperlink that points to a specific page, rather than to a top-level Web page such as a home page. In Oracle Fusion CRM, use a deep link to bypass a home page, and link directly to the details pages for specific objects. In addition, the deep link URLs that you can create are simplified to be more user-friendly.

Review these aspects of the deep linking feature before you begin to use deep links to connect to specific application pages.

- Supported objects
- Deep link URL patterns
- User authentication

**Supported Objects**

You can create deep links that connect to the details pages for certain CRM objects. The details page, which is part of an object’s work area, is the page where users can view more details about an object.

**Note**

In Oracle Fusion CRM, deep links connect to the desktop version of the details pages for the objects listed below. Deep links are not supported for use with simplified pages.

Use a deep link to connect directly to the details pages for these objects:

- Opportunity
- Lead
- Customer
- Contact

**Deep Link URL Patterns**

The deep link URL contains all the information needed to point to a particular details page. The deep link URL pattern that you must use is formatted in a user-friendly way, and subsequently translated by the Deep Link servlet in the middle tier. The Deep Link servlet reads the incoming request parameters, generates a new URL, and redirects the request to the deep-linked application.

Depending on the CRM Object in use, use the following syntax for deep link URL pattern to create a deep link to a CRM details page:

```
http://hostname:port/crmCommon/faces/deeplink?
ObjType=<object_name>&ObjId=<123456>
```

In this URL, replace `ObjType` with one of the following supported CRM objects:

- Opportunity. For this object, the deep link URL would be like `http://example-host:7100/crmCommon/faces/deeplink?ObjType=Opty&ObjId=99123456`.
- Lead. For this object, the deep link URL would be like `http://example-host:7100/crmCommon/faces/deeplink?ObjType=Lead&ObjId=99999700000`.
- Customer. For this object, the deep link URL would be like `http://example-host:7100/crmCommon/faces/deeplink?ObjType=Customer&ObjId=10000001502`. 
• Contact. For this object, the deep link URL would be like http://example-host:7100/crmCommon/faces/deeplink?ObjType=Contact&ObjId=10000001619.

User Authentication

The Deep Link servlet requires authentication, which means that a user will have to log in to gain access to that page, if not previously authenticated. After login, the user is redirected to the target deep link page. If the user is already authenticated at the time of clicking the deep link, then no login is required and the target deep link page is immediately displayed.

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.

<table>
<thead>
<tr>
<th>Source Code</th>
<th>Campaign</th>
<th>Stage</th>
<th>Audience</th>
<th>Treatment</th>
<th>Promotion</th>
<th>Coupon</th>
</tr>
</thead>
<tbody>
<tr>
<td>19999800</td>
<td>Seasonal Product Lines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19999866</td>
<td>Seasonal Product Lines</td>
<td>Summer Interaction</td>
<td>Gold Customers</td>
<td>Gold Customer Product Promotion</td>
<td>Preferred Customer</td>
<td>Limited Time Discount</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------</td>
<td>-------------------</td>
<td>----------------</td>
<td>---------------------------------</td>
<td>-------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>19999866</td>
<td>Seasonal Product Lines</td>
<td>Summer Interaction</td>
<td>Gold Customers</td>
<td>Gold Customer Product Promotion</td>
<td>Preferred Customer</td>
<td>Gold Customer Advantage</td>
</tr>
<tr>
<td>20000046</td>
<td>Seasonal Product Lines</td>
<td>Summer Interaction</td>
<td>Silver Customers</td>
<td>Silver Customer Product Promotion</td>
<td>Preferred Customer</td>
<td>Limited Time Discount</td>
</tr>
</tbody>
</table>

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.

**Resulting Task Template**

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.


**Resulting Task Template**

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.

The following table lists the stages and corresponding elements.

<table>
<thead>
<tr>
<th>Stage Type</th>
<th>Design Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound</td>
<td>Audience</td>
</tr>
<tr>
<td></td>
<td>Treatments</td>
</tr>
<tr>
<td></td>
<td>Allocations</td>
</tr>
<tr>
<td></td>
<td>Schedule</td>
</tr>
<tr>
<td></td>
<td>Additional Options</td>
</tr>
<tr>
<td>Event</td>
<td>Event Details</td>
</tr>
<tr>
<td>Advertisement</td>
<td>Treatments</td>
</tr>
<tr>
<td>Lead Generation</td>
<td>Audience</td>
</tr>
<tr>
<td></td>
<td>Additional Options</td>
</tr>
<tr>
<td></td>
<td>Lead Options</td>
</tr>
<tr>
<td>Social site</td>
<td>Event Details</td>
</tr>
</tbody>
</table>

The stage types and corresponding 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 an existing tab, or may add an additional tab to capture more stage attributes.

Outbound Stage

An Outbound stage has the following design elements:

- **Audience**

  The audience collectively describes all the contacts you want to target in the campaign. For an Outbound stage, the audience can include any combination of segment trees, segments, and lists. After the campaign is launched, you can view the audience members on the campaign members page.

- **Treatments**

  Treatment is a marketing term describing the content that is to be delivered to the target audience of a marketing campaign. You can add any number of treatments from the Treatments tab. Optionally, you can create one or more 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.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>You must allocate 100 percent of the selected audience row, whether segment or list, to a treatment.</td>
</tr>
<tr>
<td>Advanced</td>
<td>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.</td>
</tr>
</tbody>
</table>
• Activities

The Activities tab shows activities and dates for the stage.

• Schedule

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 on the date scheduled for the stage, each step immediately following completion of its predecessor.

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.

---

**Note**

Customized BPEL processes are not available to Oracle Fusion Marketing, Cloud Service customers.

---

• Additional Options

This page provides additional choices for loading campaign members.

• Monitor

This page shows details of stage execution and provides the option to re-execute a stage.

**Event Stage**

An event stage has the Event Details design element, where you can select from the list of event types, select the venue, and optionally, add attachments.

**Advertisement Stage**

An Advertisement stage has the following design elements:

• Treatments

Treatment is a marketing term describing the content that is to be delivered to the target audience of a marketing campaign. You can add any number of treatments from the Treatments tab. Optionally, you can create one or more treatments to include in your campaign.

• Monitor

This page shows details of stage execution and provides the option to re-execute a stage.
Lead Generation Stage

A Lead Generation stage has the following design elements:

- **Audience**
  The audience collectively describes all the contacts you want to target in the campaign. For a Lead Generation stage, the audience can include segment trees and segments, but not lists.

- **Additional Options**
  This page provides additional choices for loading campaign members.

- **Monitor**
  This page shows details of stage execution and provides the option to re-execute a stage.

- **Lead Options**
  After 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.

Social Stage

When you publish a social campaign in Oracle Fusion Social Marketing, a multistage campaign with a Social site stage is created automatically, to enable response tracking.

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 Oracle WebCenter Content. The ID of this text file, together with the ID of the treatment stored in Oracle WebCenter Content, 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**

**Importing Marketing Treatments: Explained**

You can create or update marketing treatments by navigating to the Treatments work area and treatment templates by navigating to Setup and Maintenance and selecting the Manage Marketing Treatment Templates task, or by importing data through interface tables. To import treatments and treatment templates, you use the tool of your preference to load the data then use scheduled processes to import the data. Having a good understanding of the application objects, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing treatments:

- Import interface tables, destination tables, and application entities
- Scheduled processes

**Import Interface Tables, Destination Tables, and Application Entities**

The treatment and treatment templates consists of many entities that form the treatment. Each entity is tied to an interface table.

**Note**

Spreadsheets containing detailed information about each interface table, including the 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 interface tables, destination tables, and resulting application entities:

<table>
<thead>
<tr>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Entities</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKT_IMP_TREATMENTS</td>
<td>MKT_TM_TREATMENTS</td>
<td>Marketing Treatment and Treatment Template</td>
</tr>
<tr>
<td>MKT_IMP_TRMT_ITEM_ASSOC</td>
<td>MKT_TM_TRMT_ITEM_ASSOC</td>
<td>Treatment and Treatment Template Product and Product Group</td>
</tr>
<tr>
<td>MKT_IMP_TRMT_PROM_ASSOC</td>
<td>MKT_TM_TRMT_PROM_ASSOC</td>
<td>Treatment Promotion</td>
</tr>
<tr>
<td>MKT_IMP_TRMT_URLS_ASSOC</td>
<td>MKT_TM_TRMT_URLS_ASSOC</td>
<td>Treatment and Treatment Template Marketing URL</td>
</tr>
</tbody>
</table>

To obtain the unique IDs of existing application data, use the Define Data Export Setup and Maintenance task list.
You can import extensible, custom attributes in the same process as your treatment object. Design your object model extensions in 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.

**Scheduled Processes**

Navigate to **Scheduled Processes** to schedule the import of data from the interface tables to the destination tables.

The following table displays the process you can schedule to import treatments and treatment templates:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Process Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BulkTreatmentsJob</td>
<td>Import Marketing Treatments</td>
</tr>
</tbody>
</table>

**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, select a local file and upload it to the server, or use the Content Library to host images in a public location. 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.

Sales campaigns have a special feature you can use to compose a personalized text message for each recipient. This feature is not available for mini campaigns or multistage campaigns.
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
- Subscribe to List
- Unsubscribe from List
- One Click Unsubscribe
- Profile Update
- E-mail as Link
- Add Address to E-mail Client

Note
Only the first three response forms are available in a sales campaign or mini 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.

By default, all standard URLs are tracked automatically. You can optionally enable tracking for Ad Hoc URLs, or disable tracking for standard URLs. With tracking enabled, 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.

Configure Marketing Campaigns and Responses

Marketing 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.

Marketing profile options are categorized into the following:

- Define Treatments
- Design Multistage Campaigns
- Execute Campaigns
- Manage Budgets
- Manage Responses
- Define File-Based Data Import
- Define Segmentation Manager

Define Treatments

The following table lists the profile options that affect Treatments:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow Treatment Template</td>
<td>Specify a default treatment template.</td>
</tr>
<tr>
<td>E-Mail Treatment Distribution Profile Default</td>
<td>Select the fulfillment supplier's distribution profile used as the default value when creating e-mail treatments.</td>
</tr>
<tr>
<td>E-Mail Treatment Supplier Default</td>
<td>Select the fulfillment supplier used as the default value when creating e-mail treatments.</td>
</tr>
<tr>
<td>Read Receipt Response Form Default</td>
<td>Enable automatic defaulting of read receipt tracking to HTML e-mail treatment content. This profile option can be updated at the user level.</td>
</tr>
<tr>
<td>SMS Treatment Distribution Profile Default</td>
<td>Select the fulfillment supplier's distribution profile used as the default value when creating short message service treatments.</td>
</tr>
<tr>
<td>SMS Treatment Supplier Default</td>
<td>Select the fulfillment supplier used as the default value when creating short message service treatments.</td>
</tr>
</tbody>
</table>
Treatment Content Directory  Specify the folder path in Oracle WebCenter Content to store marketing treatment content.

Treatment Template Directory  Specify the folder path in Oracle WebCenter Content to store marketing treatment templates.

Treatment Template Name Prefix  Specify a default prefix for treatment template names. This profile option can be updated at the user level.

Design Multistage Campaigns

The following table lists the profile options that affect multistage campaigns:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ad Hoc Campaign Approval</td>
<td>Enable the campaign owner to select the approver from a list of resources. If not enabled, the first approver is the campaign owner’s manager.</td>
</tr>
<tr>
<td>Enabled Approval Timeout Interval</td>
<td>Specify the number of days an approver has to respond before the approval request times out.</td>
</tr>
<tr>
<td>Enforce Campaign Approval Enabled</td>
<td>Enable application verification of the campaign approved status before the campaign is launched.</td>
</tr>
</tbody>
</table>

Execute Campaigns

The following table lists the execute campaigns profile options:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Campaign Lead Generation Load Format Default</td>
<td>Select the default path and name of the file containing the column mapping used to load sales leads during campaign stage execution. The default load formats are displayed in the Additional Options tab when designing a lead generation stage. This profile option can be updated at the user level. Fusion passes the campaign lead generation load format to the segmentation server when a stage is being loaded and launched.</td>
</tr>
<tr>
<td>Campaign Member Load Format Default</td>
<td>Select the default path and name of the file containing the column mapping used to load campaign members during campaign stage execution. The default load formats are displayed in the Additional Options tab when designing an interaction stage. This profile option can be updated at the user level. Fusion passes the campaign member load format to the segmentation server when a stage or mini campaign is being loaded and launched.</td>
</tr>
<tr>
<td>Recipient File Folder Default</td>
<td>Specify the Oracle WebCenter Content directory path for member list files used for marketing campaign fulfillment.</td>
</tr>
<tr>
<td>Search Marketing Campaigns Enabled</td>
<td>Enable the display of Search Marketing Campaigns, a tab in the Campaigns Overview user interface containing marketing campaigns using search engine providers.</td>
</tr>
</tbody>
</table>
Manage Budgets

The following table lists the profile options that affect the management of budgets:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Budget Fund Request and Claim 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>Budget Funding Tolerance Default</td>
<td>Specify the funding tolerance percent to default when creating budgets. The percentage is used to calculate an allowable amount that total fund requests can exceed the budgeted amount.</td>
</tr>
<tr>
<td>Claim Submission Deadline Default</td>
<td>Specify the number of days added to the budget creation date to determine the default claim submission deadline date.</td>
</tr>
<tr>
<td>Fund Request Submission Deadline Default</td>
<td>Specify the number of days added to the budget creation date to determine the default fund request submission deadline date.</td>
</tr>
</tbody>
</table>

Manage Responses

The following table lists the profile options for the management of responses:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marketing Suspect Synchronization Batch Size</td>
<td>Specify the number of rows to batch during synchronization of records between the marketing suspects table and the customer party table.</td>
</tr>
<tr>
<td>Marketing Suspect Synchronization Segment Name</td>
<td>Specify the name of the segment that identifies the marketing suspect contacts to be excluded from the customer party table during the campaign load process.</td>
</tr>
</tbody>
</table>

Define File-Based Data Import

The following table lists the profile option for define file-based data import:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Universal Content Management Applet Enabled</td>
<td>Enable Oracle WebCenter Content applet to browse and load large files when creating file import processing activities.</td>
</tr>
</tbody>
</table>

Define Segmentation Manager

The following table lists the profile option for the segmentation manager:

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segmentation Server Default</td>
<td>Specify the default segmentation server host name.</td>
</tr>
</tbody>
</table>
Manage Social Marketing Connectors

Oracle Social Marketing and Oracle Fusion Marketing: How They Work Together

Using Oracle Social Marketing, you can create social media content (web forms on Facebook pages, Twitter Tweets, and so forth) and then track visitors' responses to this content in an Oracle Fusion marketing campaign.

You can also use Oracle Fusion Marketing to create outbound e-mail campaigns that point to social media content (such as Facebook pages or tabs), and then track visitors' interactions with the content in an Oracle Fusion marketing campaign.

This topic describes the interaction between Oracle Social Marketing and Oracle Fusion Marketing.

The following diagram illustrates the interaction.

Setting Up

You must perform setup tasks on both the Oracle Social Marketing and Oracle Fusion Marketing sides before you can use Oracle Fusion Marketing.
to track responses to your Oracle Social Marketing campaigns. This provides
the information that Oracle Fusion and Oracle Social Marketing need to
communicate with each other, as well as verifying that a valid connection can be
established between them.

- In Oracle Fusion Marketing, you specify your Oracle Fusion user name
  and password, as well as the URL you use to sign in to your Oracle Fusion
  CRM account.

- In Oracle Social Marketing, you specify the location to connect to Oracle
  Fusion, as well as your Oracle Fusion user name and password.

You can perform a test to determine whether the information you provided will
result in a valid connection to Oracle Social Marketing.

**Defining Facebook Tabs**

After you have successfully connected to Oracle Social Marketing, you can
create a Facebook tab that includes one or more web forms that visitors can
complete. In Oracle Social Marketing, navigate to Forms > Fusion CRM Forms.
On the Enter Your Fusion CRM Information page, you can specify your Oracle
Fusion CRM user name and password and the URL to your Oracle Fusion
service. You click the Link to Campaign button on the Enter Your Fusion CRM
Information page to publish the campaign to Facebook and set up an Oracle
Fusion Marketing campaign to collect the responses. After the form is published
to Facebook, respondents can visit the page. Their responses when filling out
your web form are sent to your Oracle Fusion Marketing campaign.

When you publish a web form to Facebook that you have created using Oracle
Social Marketing, Oracle Fusion automatically defines a new Marketing
campaign. Marketing campaigns defined by Oracle Fusion Marketing are set
up as multistage campaigns, but only a subset of the campaign's fields are
displayed. Multistage campaign fields that are not displayed in campaigns set up
by Oracle Fusion Marketing include:

- Template
- Design Status
- Status
- Execution Status
- Enforce Approvals
- Execution Start Date
- Planned End Date

The campaign set up by Oracle Fusion contains only a single stage, of the type
Social Site. The purpose of this campaign is to collect the responses from visitors
who visit your Facebook page and fill out your web form. Responses are tracked
based on unique source codes assigned to each web form. Your Facebook tab can
contain multiple web forms, or you can have multiple Facebook tabs, but each
response includes its own unique code for tracking purposes.

If you choose, you can associate the campaign containing the Social Site stage
with another Oracle Fusion Marketing multistage campaign.
Visiting the Campaign and Viewing Responses

After your Facebook tab has been published, visitors can view it and complete the web form. When a visitor submits a web form, a response is automatically generated and included in the Fusion Marketing campaign.

When a visitor to the site provides a response such as filling out a web form, the relevant information is passed back to Oracle Fusion. Responses are captured using special tracking information that is embedded in the links that a visitor clicks to access the campaign content. Using this tracking information, Oracle Fusion can determine which visitors responded to which campaigns, and what sort of responses they made.

Marketers can go to the Campaign work area in Oracle Fusion Marketing and view the responses in the Results area of the social campaign. The Responses tab shows all responses to the campaign, including the date and the name of the respondent.

Customizing E-Mail and Web Pages for Outbound Campaigns

When creating e-mail or Web pages as part of an Oracle Fusion outbound campaign, you can include links to Facebook pages that include your content, or provide links that allow your recipients to send your content to their contacts using Twitter.

Responses to Outbound Campaigns

When you include links to Facebook pages, Fusion Marketing tracks when the recipient clicks on the link in your e-mail or on your Web page and visits your page. Oracle Fusion Marketing also tracks visitor response if a recipient fills out a Web form that you have created using Oracle Social Marketing.

When you include a preconfigured Twitter Tweet, recipients who click the link are sent to their Twitter home page, where they can sign in and view your preconfigured Tweet content. They have the option to edit the preconfigured content or to send the Tweet as is to their Twitter feed.

In either case, Oracle Fusion Marketing automatically tracks the responses and, as is the case with responses to web forms, includes the response information in the Results area of the social campaign. Marketers can visit the campaign's Responses tab to view the results.
Define Campaign Fulfillment System

Importing Marketing Suppliers: Explained

You can create or update marketing suppliers by navigating to **Setup and Maintenance** and selecting the **Manage Marketing Supplier Details and Distribution Profiles** task or by importing data through interface tables. To import suppliers, you use the tool of your preference to load the data then use scheduled processes to import the data. Having a good understanding of the application object, interface tables, and destination tables will help you prepare your import data.

Consider the following when importing marketing suppliers:

- Import interface tables, destination tables, and application entities
- Scheduled processes

**Import Interface Tables, Destination Tables, and Application Entities**

The marketing supplier consists of many entities that form the supplier. Each entity is tied to an interface table.

**Note**

Spreadsheets containing detailed information about each interface table, including the 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 interface tables, destination tables, and resulting application entities:

<table>
<thead>
<tr>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Entities</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKT_IMP_VENDORS</td>
<td>MKT_TM_VENDORS</td>
<td>Marketing Supplier</td>
</tr>
<tr>
<td>MKT_IMP_VENDOR_PROFILES</td>
<td>MKT_TM_DLV_PROFILE</td>
<td>Distribution Profile and Delivery</td>
</tr>
<tr>
<td></td>
<td>MKT_TM_DLV_PRF_PARAMS</td>
<td>Channel Parameters</td>
</tr>
</tbody>
</table>
To obtain the unique IDs of existing application data, use the Define Data Export Setup and Maintenance task list.

**Scheduled Processes**

Navigate to **Scheduled Processes** to schedule the import of data from the interface tables to the destination tables.

The following table displays the process you can schedule to import marketing suppliers:

<table>
<thead>
<tr>
<th>Process Name</th>
<th>Process Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BulkVendorJob</td>
<td>Import Marketing Suppliers</td>
</tr>
</tbody>
</table>
E-Mail Architecture for Marketing: How It Fits Together

E-mail marketing includes a combination of components designed to support high volume, personalized e-mail messages and track e-mail bounces and click-through responses. This topic provides an overview of the following e-mail marketing installation components and how they fit together:

- Business Intelligence and Oracle WebCenter Content
- E-Mail Sending Daemon (ESD) and the Mail Transfer Agent (MTA)
- Bounce Handling Daemon (BHD)
- Click-Through Daemon (CTD)

The following figure displays an example of a typical architectural structure for e-mail marketing server components:
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. Oracle Marketing Segmentation is fully integrated with Oracle Business Intelligence and uses the same metadata as the Business Intelligence reporting tools. When an e-mail campaign is launched, the contact details are collected using the information in the Business Intelligence Repository, formatted using the Business Intelligence server, and then stored in Oracle WebCenter Content. Oracle WebCenter Content also stores the campaign e-mail content, headers, footers, and attachments specified during campaign design.

**E-Mail Sending Daemon and the Mail Transfer Agent**

The e-mail sending daemon assembles each outbound e-mail message for a campaign. It then sends each message to the outbound mail transfer agent (MTA) for delivery.

The e-mail sending daemon listens for SOAP requests from the marketing E-Mail Server. A SOAP request includes the required file names found in Oracle WebCenter Content. The e-mail sending daemon must be able to communicate
with one or more outbound mail transfer agents to send mailings over the Internet. The e-mail sending daemon must also be able to inform the marketing E-Mail Server when it has completed as well as details of synchronous e-mail bounces for delivery errors that occurred while it communicated using simple mail transfer protocol (SMTP) to the mail transfer agents.

The most common placement for the e-mail sending daemon is within the corporate network, behind the DMZ. However, the e-mail sending daemon can be placed inside the DMZ or outside the firewall, if there is a port opened to connect to the marketing E-Mail Server and Oracle WebCenter Content.

**Bounce Handling Daemon**

The bounce handling daemon tracks e-mail messages that cannot be delivered, parses the returned e-mail messages, and records the cause of the e-mail bounce.

It is recommended that you place the bounce handling daemon in the DMZ. However, you can place the bounce handling daemon behind an inbound mail transfer agent. The approach that you choose depends on the configuration of your network, DMZ, existing inbound mail transfer agent, and firewall.

**Click-Through Daemon**

The click-through daemon tracks clicks made by the e-mail recipient on trackable URLs included in the e-mail content. It listens for HTTP requests, such as forward to a friend, subscribe to list, and one click unsubscribe.

You can place the click-through daemon in the DMZ, inside or outside the firewall, if a port is opened that allows it to connect to the marketing E-Mail Server using SOAP. Web proxy servers can be used to route the HTTP requests to the click-through daemon server.

**Manage Marketing E-Mail Sending Daemon: Explained**

The E-Mail Marketing Server is a combination of components designed to support high volume, personalized e-mail messages, and to track e-mail bounces and click-through responses. The e-mail sending daemon is the outbound component for campaign e-mail messages.

Installing the e-mail sending daemon is part of the provisioning process. For more information on provisioning, see the Oracle Fusion Applications Quick Installation Guide.

This topic includes the following:

- Outbound mail transfer agents
- Universal Content Management connection
- Volume thresholds and throughput
- Spam scoring server settings
- E-mail sending daemon properties file
Outbound Mail Transfer Agents

The e-mail sending daemon (ESD) needs one or more outbound mail transfer agents (MTAs) to deliver e-mail over the Internet. You specify the host and port numbers of your MTAs in the ESD properties file. If you specify multiple MTAs, then the e-mail sending daemon uses the additional MTAs as backup. For example, when the e-mail sending daemon starts, it uses the first MTA listed in the properties file. If the MTA fails, the next MTA listed is used. To use the first MTA again, you must restart the e-mail sending daemon, usually by rebooting the server.

Universal Content Management

The e-mail sending daemon listens for SOAP requests from the marketing E-Mail Server. A SOAP request includes the required file names found in Universal Content Management. You specify access information for Universal Content Management in the ESD properties file.

Volume Thresholds and Throughput

You can specify or adjust the threshold value property that determines when the campaign priority needs to be honored. When the threshold is reached, the priority value specified during campaign design is used. Processing for campaigns with less priority is temporarily paused until the processing for higher priority campaigns is completed.

You can limit the maximum throughput rate for the e-mail sending daemon, specified in messages per second.

- A value of -1 is highest throughput rate possible and is the default setting.
- A value of 250-350 supports a medium level of throughput.
- A value of 250-50 supports a low level of throughput.

Spam Scoring Server Settings

You can install SpamAssassin, an open-source spam scoring software provided by Apache Software Foundation, to assist marketers in creating campaign content that is less likely to be identified as junk mail by the recipients e-mail software. Once you have completed the installation steps provided by Apache Software Foundation, update the e-mail sending daemon properties file with the software location and to enable the reporting of a spam score and score reasons when testing e-mail campaign content.

ESD Properties File

You can modify the e-mail sending daemon properties file if changes are required after installation.

The following table describes the e-mail sending daemon properties file (esd.properties):
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>esd.mailServers</td>
<td>The MTA server, including host name and port number, used by the ESD to deliver e-mails. When listing more than one MTA, use a comma between each MTA.</td>
</tr>
<tr>
<td>esd.contentHandler.server</td>
<td>The Universal Content Management URL used to access the content server.</td>
</tr>
<tr>
<td>esd.mountPoint</td>
<td>The absolute path of the file directory where the ESD writes the content files.</td>
</tr>
<tr>
<td>esd.handler.bounce.webservice.url</td>
<td>The web service URL used by the ESD to log a bounced e-mail message. To locate this web service, search for <code>MktDialogsCampMemberUpdateComposite</code> composite deployed within the Marketing SOA environment.</td>
</tr>
<tr>
<td>esd.handler.updateStatus.webservice.url</td>
<td>The web service URL used by the ESD to log a completed process. To locate this web service, search for <code>MktDialogsCampFulfillmentStatusComposite</code> composite deployed within Marketing SOA environment.</td>
</tr>
<tr>
<td>esd.priorityThreshold</td>
<td>The threshold that determines when the campaign priority needs to be honored.</td>
</tr>
<tr>
<td>esd.smtp.throttleVal</td>
<td>This property will determine the e-mail throughput rate for the system.</td>
</tr>
<tr>
<td>esd.spamPort</td>
<td>The port number of the server where SpamAssassin is installed. The default port number is 783.</td>
</tr>
<tr>
<td>esd.spamServer</td>
<td>The host name or IP address of the server where SpamAssassin is installed.</td>
</tr>
<tr>
<td>esd.reportSpamErrors</td>
<td>The Boolean value used to enable the display of SpamAssassin scoring when previewing marketing e-mails.</td>
</tr>
</tbody>
</table>
Oracle Fusion Marketing Segmentation: Architectural Overview

Oracle Fusion Marketing Segmentation is fully integrated with Oracle Business Intelligence Enterprise Edition (OBIEE) and uses the same metadata as the Oracle Business Intelligence Applications (OBIA) reporting tools.

**Oracle Fusion Marketing Segmentation Architecture**

You interact with information in Oracle Business Intelligence Applications (OBIA) and Oracle Transactional Business Intelligence (OTBI) using Oracle BI (Business Intelligence) Enterprise Edition components. The following figure outlines the marketing segmentation architecture.
The OBIA warehouse is a unified data repository for all customer-centric data and is used to support the analytical requirements of Oracle Business Intelligence Applications.

OTBI is real-time self service component reporting directly off Fusion application data where you can create your own reports using pre-packaged functional metadata. OTBI is co-located and semantically integrated with the data warehousing metadata and can read security, flex fields and trees from the Oracle Fusion CRM Marketing application.

The RPD layer is the repository file that stores metadata. In order to use any BI tool that depends on its own metadata layer, such as BI Answers and BI Dashboards, the metadata repository must describe how queries should be constructed against the relational data sources. RPD files are used to define the physical layer.

The BI presentation layer stores the Segment, Segment Tree and List Format definitions in the Web Catalog. The catalog stores content created with the Segment Designer, Segment Tree Designer, and List Format designer. Content can be organized into folders, which are either shared or private. Types of content that can be stored in the Web Catalog include segments, segment trees, list formats, as well as reports, filters, and dashboards created with Oracle BI Answers.

**Oracle Fusion Marketing Segmentation: Implementation Planning**

This topic lists some of the considerations to take into account when planning your implementation of Oracle Fusion Marketing Segmentation.

You need to manage the necessary configuration, integration, and metadata settings to fully enable marketing segmentation functionality. Segmentation setup and configuration is done in two parts.

**Define Segmentation Manager**

The first part of the configuration happens when the OBIEE server is being implemented and configured. These tasks are outlined as follows:

- **Manage Segmentation Manager Configuration Parameters**
  
  Configure deployment parameters, default values, and connection pools to the database server and list output to database tables for the underlying business intelligence infrastructure.

- **Manage Segmentation Manager Enterprise Content Management Integration**

  You can configure deployment and connection parameters for Oracle Enterprise Content Management server repository using Oracle Enterprise Manager. You do not need to configure any connections for Fusion Application users. When campaigns are launched, the target audience contact details are collected using the information in the Business Intelligence Repository, formatted using the Business Intelligence server, and then stored in a content server. You, as the
administrator, can configure for connections to the Oracle Content Server for non Fusion application users.

- **Manage Segmentation Manager Metadata for Marketing**

  Create and maintain segmentation metadata, such as target levels, presentation catalog subject areas, and schema mappings, used when creating segments, segment trees, and list formats.

**Marketing Operations Manager Role**

The second part of the configuration is done by a Marketing Operations Manager role within the Oracle Fusion Marketing Segmentation application.

- **Manage List Formats**

  Marketing Segmentation comes with pre-seeded list formats to output segments for campaign execution, e-mail personalization and lead creation. If there is a need to output a segment in a different format, such as including different or custom attributes in the output, then a new List Format can be created to accommodate that need.

- **Manage Segmentation Manager Contact Planning Rules**

  Define frequency and volume constraints that can be applied globally or against a specific customer communication channel. Use segment inclusion criteria to define the conditions where the rules apply. Allow a company to place volume, frequency and recency rules on all forms of communication across customer, channel and program dimensions. Define a set of segments which represent the segments of the population that will be included. For example, a segment which includes all the customers that are opted into e-mail.

**Marketing Segmentation Administration: 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
- Security Mappings for Fusion Users of Marketing Segmentation


**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. Refer to the Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.

**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. Refer to the Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.
- To provide default campaign load and data load formats for campaigns, navigate to Setup and Maintenance 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, refer to 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. Refer to the Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.

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 segmentation metadata, see the chapter on Configuring Marketing Segmentation Metadata, and the appendix on Marketing Segmentation Metadata Content in the Marketing Segmentation Guide. Refer to the Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.

Security Mappings for Fusion Users of Marketing Segmentation

- For information about the security mappings and access levels specific for Fusion users of Marketing Segmentation, see the appendix on Security Mappings for Fusion Users of Marketing Segmentation in the Marketing Segmentation Guide. Refer to the Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.

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.

Oracle Marketing Segmentation Guide describes how to create segments and segment trees.

Manage Marketing Segments and Segment Trees

- For information about marketing segments and segment trees, see the Using Marketing Segments and Segment Trees chapter in the Marketing Segmentation Guide. Refer to Oracle Marketing Segmentation Guide on Oracle Technology Network at http://www.oracle.com/technetwork.
Configuring Custom Subject Areas for Segmentation: Points to Consider

This topic outlines the setup required to target some organization contacts using criteria from a custom object that has a relationship with an Organization Customer (B2B). It also specifies some setup variations for configuring custom objects related to B2C Customers (Consumers) and custom objects related to contacts.

Configuring Custom Subject Areas in Segments

The following example is used to illustrate your task. You are a B2B Company and want to target Organization Contacts in a marketing campaign. The contacts work for Organizations that have placed an order with your company in the past month.

Before You Begin

The following lists the tasks that you must perform before you begin to configure the custom subject area for segmentation:

- Create a custom object named Orders and include custom fields such as date, amount, product and so on using the dynamic choice list. Select Trading Community Organization Profile as the target object for the dynamic choice list to allow you to select the customer for which each new order is required. This action creates a relationship between Order and Organization Customer.

- Create a work area for the custom Order object and expose the necessary fields in the Overview, Create and Detail pages.

- Create data records for the Order object either through the UI, through Import or through web services. When creating records, make sure that you select the Organization Customer for which each Order is required.

- Create a custom subject area for the Order object and expose that subject area for segmentation. When selecting fields to include in the custom subject area you must choose the Party ID from the Trading Community Organization Profile object.

Custom Subject Area Segment Configuration

Do the following to configure the custom subject area for segmentation:

1. Associate the subject area to one or multiple Target Levels.

   In this example, the custom object has a relationship to Organization Customer only, so it must be related to the Real Time Customers target level only. The Qualifying Identifier must be set to the same level as the target level (Customers Real Time) and the Mapped Field must be set to
the Party ID of the Trading Community Organization Profile object. This association ensures that Marketing Segmentation uses the Organization Customer Party ID to identify and count individual customers.


Set the Qualifying Identifier to the same level as the target level (Real Time Customers) and set the mapped field to the Party ID of the Trading Community Organization Profile object. This association ensures that Marketing Segmentation uses the Organization Customer Party ID as the input parameter when querying the database for the set of customer data to include in the segment.

Using Custom Subject Areas in Segments

Now that the custom subject area has been set up and Marketing Segmentation has been configured, you can now use the custom subject area when creating segments. Using our example, the target for the segment is individual contacts that work for organizations which have an order. You need to create a segment that targets Real Time Customers and then nest (relate) the Real Time Contacts subject area.

Perform the following steps:

1. Navigate to the Audience work area and select Create Segment. Enter a name for your segment, select the Real Time Contacts target level, and click Save and Design. The Segmentation UI will be displayed.

2. Click the Select Another Target Level icon in the upper right corner (next to the Save icon) and select the Real Time Customers target level.

3. Select the Add/Remove Subject Areas icon in the upper left corner of the screen (in the Subject Area section next to Refresh icon). Select the new custom Orders subject area that you previously created.

4. Create the segment criteria for the set of Organizations that you want to target (based on orders, or customer attributes), then save the segment.

Note

This segment will not appear in Marketing Segmentation UI at any time. It is a nested segment used by the Contacts Real Time segment that is being created.

5. Once the Customer level segment is saved, click Go back to segment_name link in the upper left of the Segment Designer section. The segment_name is the name of the segment that you are designing. This action returns to the original segment that is at the Contact level and the Customer level segment will be nested as the first criteria.

6. Add any other criteria necessary to the segment, can be at the customer or contact level (customer name, contact e-mail, and so on).

7. Save the segment and update counts

The counts should reflect the number of contacts that are related to the companies which meet the Orders criteria (plus any other criteria input into the segment).
Custom Objects Related to B2C Customers (Consumers)

The set up and process for creating the custom subject area for custom objects related to Consumers is almost identical to the process outlined in the example above with a few exceptions as follows:

- When creating the custom object and inserting the dynamic choice list, the target object must be the Trading Community Person Profile object. This object relates the Person Customer to the custom object (instead of an Organization Customer).

- When setting up the custom subject area, the Party ID of the Trading Community Person Profile object must be added to the subject area in the Field selection step.

- When configuring segmentation on the custom subject area, the target level should be set to Real Time Consumers. The Qualifying Identifier must be set to Consumers Real Time and the Party ID from the Trading Community Person Profile object must be used as the mapped field in both steps.

- A nested segment is not needed when you create a segment. The custom subject area is related directly to the person customer (consumer) so that object can be used in a segment that targeting consumers.

Custom Objects Related to Contacts

The set up and process for creating the custom subject area for custom objects related to Contacts is almost identical to the process outlined in the example above with a few exceptions as follows:

- When creating the custom object and inserting the dynamic choice list, the target object must be the Trading Community Customer Contact Profile object. This object relates the Contact to the custom object (instead of an Organization Customer or Person Customer).

- When setting up the custom subject area, the Party ID of the Trading Community Customer Contact Profile object must be added to the subject area in the Field selection step.

- When configuring segmentation on the custom subject area, the target level should be set to Real Time Contacts. The Qualifying Identifier must be set to Contacts Real Time and the Party ID from the Trading Community Customer Contact Profile object must be used as the mapped field in both steps.

- A nested segment is not needed when you create a segment. The custom subject area is related directly to the person customer (consumer) so that object can be used in a segment that is targeting consumers.
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. Refer to the Oracle Fusion Transactional Business Intelligence Administrator’s Guide.

Manage Security for Transactional Business Intelligence

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>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</td>
<td>View Manager Hierarchy</td>
</tr>
<tr>
<td></td>
<td>Participant Manager</td>
<td></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>

**Summary of the Tasks**

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. On the Overview page of the Setup and Maintenance work area, click the All Tasks tab.
2. 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>
3. Click Search.
4. 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 described in the Oracle Fusion Applications Administrator's Guide and Oracle Fusion Applications Extensibility Guide for Developers. As you read content from these guides, note that the guides mention 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.

Note
Oracle Enterprise Manager Fusion Applications Control is not available for Oracle Cloud implementations.

Selecting the Appropriate Implementation Task

Each Enterprise Scheduler job definition task uses one Java EE application, which is referenced in the task name. You must use the right task because, to access the product job definition that you want to view or work on, the view objects must be contained in the application. If you do not select the right task, then the job definition will not be displayed properly or function correctly. The application name is usually the same as the product that the job definition belongs to, but not always.

- For example, the Oracle Fusion Payables Java EE application contains the Oracle Fusion Expenses product view objects. To create or maintain a job definition for use in Expenses, you select the Manage Custom Enterprise Scheduler Jobs for Payables and Related Applications task.
- In another example, the Oracle Fusion Payments product view objects are contained in both Oracle Fusion Payables and Oracle Fusion Receivables...
Java EE applications. You need to select the task appropriate to the job definition for Payments. Use the Manage Custom Enterprise Scheduler Jobs for Receivables and Related Applications task if the job is for receivables functionality, or the Manage Custom Enterprise Scheduler Jobs for Payables and Related Applications task if the job is for payables functionality.

- Use the task description to see the products that correspond to the Java EE application specified in the task name. For example, the description for the Payables task lists Oracle Fusion Payables, Assets, Expenses, and Payments.

- You can view task descriptions in the help window for the task, if any, or in the generated setup task lists and tasks report from the Getting Started page.

- If you have access to the Manage Task Lists and Tasks page, you can also open the details for specific tasks to see the description.

- For general information about product and Java EE application relationships, use Oracle Enterprise Manager Fusion Applications Control (Fusion Applications Control).

See: Topology Section

**Viewing, Creating, and Editing Job Definitions**

- You can access predefined and custom job definitions. In the table on the Manage Job Definitions tab, the Name column displays an asterisk for predefined job definitions. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Viewing Job Definitions

- You or a technical administrator can create jobs based on Java, PL/ SQL, Oracle Business Intelligence (BI) Publisher, or any other supported technology. Every predefined or custom job must have a job definition. For Oracle Cloud implementations, custom job definitions can be created only for custom jobs based on Oracle BI Publisher reports. Refer to the Oracle Fusion Applications Administrator's Guide.

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.

- You can edit all aspects of custom job definitions. For predefined job definitions, you can't update parameters, but you can determine if user properties are read-only or not. You can also edit certain aspects of predefined definitions, which are described as job properties in the Oracle Fusion Applications Extensibility Guide for Developers.
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 application defined properties of job definitions. For example, you can use a source of country values for a Country job parameter.

Note

Since parameters for predefined job definitions cannot be edited, list of values sources are only for parameters in custom job definitions.

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

Managing Job Sets: Highlights

Oracle Enterprise Scheduler job set definitions contain metadata that identify the jobs to be included in a single submission. Instead of submitting jobs separately to process data, and in some cases, generate output, users can submit one job set that contains the desired jobs. The job set definition also determines if the jobs run in serial or parallel, or based on other predetermined logic.

A job set can contain any number of individual jobs as well as other job sets. There can also be multiple levels of nested job sets within a single job set. For example, a job set can include three jobs and two job sets, one of which contains another job set.

Managing job sets is fully described in the Oracle Fusion Applications Administrator's Guide.

Creating and Editing Job Sets

- The Oracle Fusion Applications Administrator's Guide describes accessing job sets from Oracle Enterprise Manager Fusion Applications
Control. You can also access job sets by starting in the Setup and Maintenance Overview page and using the Enterprise Scheduler job tasks for your applications.

See: Creating and Editing Job Sets

**Contextual Addresses**

**Setting Up the Mapping Service for Contextual Addresses: Points to Consider**

A contextual address is marked with an orange square contextual action icon that can be clicked to display the address on a map. Use the Mapping Service for Contextual Addresses profile option to specify the web mapping service to be used to display the map. In the Setup and Maintenance work area, go to the Manage Administrator Profile Values task to set the profile option value.

**Profile Option Default**

By default, the Mapping Service for Contextual Addresses profile option has no value. Until you enter a valid value for this profile option, users get an error when they try to open a map for any contextual address.

**Profile Option Value**

After you find and select the Use the Mapping Service for Contextual Addresses profile option, enter a mapping service URL in the Profile Value column, for example:

- `http://maps.yahoo.com/maps_result.php?q1=
- `http://bing.com/maps/?v=2&encType=1&where1=

Optionally, add parameters to the URL. For example, to avoid a locator box in Google Maps, add `&iwloc=&` to the URL, so that you enter `http://maps.google.com/maps?iwloc=&output=embed&q=` as the profile value.

**Customizations and Sandboxes**

**Supported Application Composer Customizations and Best Practices: Explained**

You can export and import supported customizations across Oracle Fusion applications that are on the same release and same patch level.
Use Customization Set Migration (CSM) to export and import supported customizations. To access CSM, select Customization Migration from the Tools section in the Navigator menu in the source or target environment.

This topic explains the following:

- Supported customizations
- Unsupported customizations
- Best practices while using CSM

**Supported Customizations**

The supported customizations for the Application Composer include:

- Object UI extensions
- Object server scripts
- Saved searches
- Workflows
- Global functions
- Object model extensions
- Relationships
- Role security privileges to access objects.
- E-mail templates: Recreate manually in the target Oracle Fusion application.
- Custom subject areas: Republish all custom subject areas in the target Oracle Fusion application. See the "Publishing Custom Subject Areas: Explained" section in the Oracle Sales Cloud Extensibility Guide.
- FND lookups
- All reports, analyses, and dashboards
- Migrate using FSM tasks: Manage Standard Lookups, Manage Custom Lookups, and Manage Set-Enabled Lookups.

**Unsupported Customizations**

Customizations that are not supported for the Application Composer include:

- Import and export artifacts generated for custom objects and fields: Regenerate manually using the Import and Export menu option in the Application Composer.

Customizations made outside the Application Composer and not supported include:

- Sales Prediction Engine (SPE) business rules: Recreate manually in the target Oracle Fusion application.
- Security job roles and duty roles: Recreate manually in the target Oracle Fusion application.

**Note**
Application Composer supports Page Composer customizations.

You must create unsupported customizations manually in the target application.

**Best Practices for Using CSM**

Some of the best practices for using CSM are:

- Do not manually create supported customizations in the target.
- Manually create customizations which are partially supported or not supported by CSM (See list above).
- Use CSM to migrate FND Lookups.
- Do not modify metadata extract.
- Migrate metadata from one instance to another.
- Import the most recent extract.
- Manually re-key security changes made in Authorization Policy Manager (APM) from source to target prior to using CSM.
- During an export or import, it is recommended that you do not make customization changes in the source or target instance.

**Exporting and Importing Supported CRM Customizations: Explained**

Once you have completed creating customizations in Oracle Fusion CRM Application Composer, you can export and import these customizations across Oracle Fusion application instances on the same release. Use Customization Set Migration (CSM) to export and import customizations.

Some of the supported customizations that you can export and import include object UI extensions, object server scripts, saved searches, workflows, global functions and so on. Do not create these supported customizations manually in the target application instance. Import these supported customizations from the source instance only.

**Note**

In the target application instance, you must create only unsupported customizations.

In order to view the customizations in the target application instance, you must first export your customizations from the source instance as compressed files in either .jar or .zip format. You can then import these compressed files in the target instance.

The following sections provide an overview of exporting and importing customizations. For more information on using CSM, see "Using Customization Set Migration to Move Customizations" in the Oracle Fusion Applications Extensibility Guide for Business Analysts on Oracle Technology Network at http://www.oracle.com/technetwork/indexes/documentation.
Note

In CSM, the changes are directly copied to the mainline and there is no sandbox involved.

Exporting Customizations

You can export customizations by creating a customization set using CSM as compressed files in either .jar or .zip format.

When exporting, keep in mind the following points:

- Do not modify the compressed files.
- Do not move customizations from the target application instance back to the source application instance.

To export customizations, do the following:

1. In the target application instance, click Navigator - Tools - Customization Migration.
   
   The Customization Migration page opens.
2. Click the Outgoing tab.
3. Create a customization set and enter the details in the Create Customization Set window.
4. Once the customization set is created, click Download.
5. Download the customization set to a local folder.


Importing Customizations

Once you have successfully exported the customizations to a compressed file format, you can import this compressed file into the target application instance.

To import customizations, use the following steps:

1. In the target application instance, click Navigator - Tools - Customization Migration.

   The Customization Migration page opens.
2. Click the Incoming tab.
3. Enter the name of the customization set.
4. Click Apply.

If you encounter any errors during the import process, (for example, the connection to the database is lost or certain process issues occur during migrating security policies), then all changes roll back automatically, reverting the target application instance to its version before the import.

If you have made any security changes in the source instance outside of CRM Application Composer, ensure that you manually re-key these security changes in the target instance prior to using CSM for importing. For example: If you have set up a custom security role in the source instance, ensure that you manually add this security role in the target instance, prior to using CSM.

**Note**

Importing earlier versions of compressed files does not roll back changes in the target instance to an earlier version of customization.

When importing, if you upload a file in a format other than *.zip or *.jar then no warning message appears, but the import job is processed and fails with an error status.

---

**Migrating FND Lookups**

You can use CSM to migrate FND lookups.

---

**Marketing Customizations**

**Customizing Sales Campaigns in Oracle Fusion Marketing: Explained**

Use the Oracle Fusion Application Composer to create custom fields for the sales campaign object, which you then add for display in the Sales Campaign work area. This topic describes the customization options for sales campaigns. Note that the fields and sections modified apply to mini campaigns as well as sales campaigns. To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

**Extensibility: Quick Steps**

This section provides a brief overview of the steps to customize the sales campaign work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then select Sales Campaign.
2. Choose the Fields node to create custom fields, or modify standard fields.
3. Choose the Pages node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Sales Campaign work area to view your changes.
Understanding Which Oracle Fusion Sales Campaign Pages Are Extensible

You can extend only the region at the top of the Sales Campaign page. The panel will adjust based on how many fields you add. The name of the region changes depending on where you are in the sales campaign flow.

Depending on the sales campaign status, the fields that you add to this region might or might not be editable. The following list shows which regions and pages are extensible, along with some notes for each:

<table>
<thead>
<tr>
<th>Region and Page</th>
<th>Are the fields editable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Campaign Summary region on the Create Sales Campaign: Enter Campaign Details page</td>
<td>Yes</td>
</tr>
<tr>
<td>Campaign Summary regions on other Sales Campaign pages</td>
<td>Only if the campaign is in Draft status. Otherwise, not editable.</td>
</tr>
<tr>
<td>Campaign Details region (displays on Sales Campaign pages for completed campaigns)</td>
<td>No</td>
</tr>
</tbody>
</table>

The following screen shot shows some of the page sections that can be customized. The regions that you can customize are indicated by red boxes.

Adding or Modifying Object Fields

To add custom fields to the Sales Campaign regions listed above, first create your custom fields by selecting the Fields node under the sales campaign object. Then, select either a standard field to modify, or create a custom field.
For standard fields, you can modify these options:

- Display Label
- Required check box
- Short Description
- Updateable
- Depends On

**Adding Your Changes to the Run Time Application**

After you’ve created custom fields for the sales campaign object, use the Application Composer’s configuration pages to add those custom fields to the extensible regions listed above. You access the configuration pages in the Application Composer from the Pages node under the Sales Campaign object. To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Sales Campaign object.
3. Select the Pages node.
4. On the Enterprise Pages tab, select the configuration page hyperlink related to the Sales Campaign page that you want to customize.
Notes About Sales Campaign Extensibility

The following customization options are not supported for sales campaigns:

- Import and export (you cannot import sales campaigns)
- Object workflow
- E-mail templates
- Business processes

The following customization options do not apply to sales campaigns:

- Role security
- Custom subject area.

Customizing Marketing Campaign Stage Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Campaign Stages object, which you then add for display in the Campaigns work area. This topic describes the extensible options for marketing campaign stages.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Campaign Stages work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Campaign Stages node.

   **Note**

   If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the Fields node to create custom fields, or modify standard fields.
3. Choose the Pages node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Campaign Stages work area to view your changes.

Understanding Which Oracle Fusion Marketing Campaign Stage Pages Are Extensible

To customize pages for marketing campaign stages, you need to know which pages and regions are extensible, and which objects to select in the Application
Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing campaign stage pages that you can access in Application Composer. To create custom fields that you can later add to any of these regions, you must select the Marketing Campaign Stages object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Campaign Stages Configuration</th>
<th>Marketing Campaign Stages Ui</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Creation Page</td>
<td>Campaigns, Create Stage, Overview region</td>
</tr>
<tr>
<td>Edit Summary Form</td>
<td>Campaigns, Edit Stage details region</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing Campaign Stages page by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**

To make minor user interface changes to Campaign Stages pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the marketing campaign stages regions listed above, first create your custom fields by selecting the Fields node under the Marketing Campaign Stages object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Campaign Stage regions listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired campaign stage regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Campaign Stages object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Campaign Stages object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Campaign Stages region that you want to customize.
Customizing Marketing Campaign Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Campaign object, which you then add for display in the Campaigns work area. This topic describes the extensible options for marketing multistage campaigns.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Campaigns work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Campaign node.

   Note

   If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

   2. Choose the Fields node to create custom fields, or modify standard fields.

   3. Choose the Pages node to show or hide fields, either custom or standard.

   4. Test your work by navigating to the Campaigns work area to view your changes.

Understanding Which Oracle Fusion Marketing Campaign Pages Are Extensible

To customize marketing campaign pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing multistage campaign pages and regions, and the related objects that you can customize using Application Composer. To create custom fields that you can later add to any of these tabs, you must select the Marketing Campaign object in the Application Composer and create your custom fields.
Next, expose those custom fields on the Manage Campaign page by accessing the appropriate Application Composer configuration page, listed in the following section.

**Note**

To make only minor user interface changes to Marketing Campaign pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the marketing campaign regions listed above, first create your custom fields by selecting the Fields node under the Marketing Campaign object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Campaigns region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Campaigns regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Campaign object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Campaign object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Campaigns region that you want to customize.

**Customizing Marketing Campaign Member Pages Using Application Composer: Explained**

Use the Application Composer to create custom fields for the Marketing Campaign Member object, which you then add for display in the Campaigns work area. This topic describes the extensible options for marketing campaign members.
To access the Application Composer, select **Application Composer** from the **Navigator** menu, under the **Tools** category.

**Extensibility: Quick Steps**

This section provides a brief overview of the steps to customize the Campaign Member work area.

Follow these steps:

1. Navigate to the **Application Composer**, select **Marketing** as the Application, expand the **Standard Objects** tree node, then expand the **Marketing Campaign Member** node.

   **Note**

   If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the **Fields** node to create custom fields, or modify standard fields.
3. Choose the **Pages** node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Campaign Member work area to view your changes.

**Understanding Which Oracle Fusion Marketing Campaign Member Pages Are Extensible**

To customize marketing campaign member pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing campaign member pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these regions, you must select the Marketing Campaign Member object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Campaign Member Configuration</th>
<th>Marketing Campaign Member UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Campaigns, Results tab, Execution Details, Campaign Members</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing Campaign Member page by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**
To make only minor user interface changes to Campaign Member pages without creating new objects or fields, use Page Composer instead of the Application Composer.

### Adding or Modifying Fields

To add custom fields to the marketing campaign member regions listed above, first create your custom fields by selecting the Fields node under the Marketing Campaign Member object. Then, select either a standard field to modify, or create a custom field.

### Adding Your Changes to the Runtime Application

To add custom fields to the Campaign Member region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Campaign Member regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Campaign Member object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Campaign Member object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Campaign Member region that you want to customize.

### Customizing Marketing List Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing List object, which you then add for display in the Audience work area. Marketing List does not have any child objects. This topic describes the extensible options for marketing lists.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

### Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the audience work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing List node.

---

**Note**
If you can't see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the **Fields** node to create custom fields, or modify standard fields.
3. Choose the **Pages** node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Audience work area to view your changes.

**Understanding Which Oracle Fusion Marketing List Pages Are Extensible**

To customize marketing list pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists Marketing List pages and regions that you can access in the Application Composer to customize those pages. To create custom fields that you can later add to any of these pages, you must select the Marketing List object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing List Configuration</th>
<th>Audience Work Area UI</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit Summary Table</strong> overview page</td>
<td>Audience, Overview, Lists tab, List table details</td>
</tr>
<tr>
<td><strong>Edit Summary Form</strong> details page</td>
<td>Audience, Overview, Lists tab, Create/Edit List, Summary region</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing List page by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**

To make only minor user interface changes to Marketing List pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the Audience regions listed above, first create your custom fields by selecting the Fields node under the Marketing List object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Audience regions listed in the table above, first create your custom fields using the Application Composer. Next, use the Application
Composer’s configuration pages to add those custom fields to the desired Audience List regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing List object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing List object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Audience work area that you want to customize.

Customizing Marketing Treatment Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Treatment object, which you then add for display in the Treatments work area. This topic describes the extensible options for marketing treatments.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Treatment work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Treatment node.

   Note

   If you can't see the choice list of applications, make sure the regional area of the page is exposed.

   2. Choose the Fields node to create custom fields, or modify standard fields.
   3. Choose the Pages node to show or hide fields, either custom or standard.
   4. Test your work by navigating to the Treatment work area to view your changes.

Understanding Which Oracle Fusion Marketing Treatment Pages Are Extensible

To customize marketing treatment pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.
For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing treatment pages and regions, and the related objects that you can access in the Application Composer to customize those pages. To create custom fields that you can later add to any of these tabs, you must select the Marketing Treatment object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Treatment Configuration</th>
<th>Marketing Treatment UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Treatments, Overview, Treatment list table</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Treatments, Overview, Create Treatment Treatment Information region</td>
</tr>
<tr>
<td></td>
<td>Treatments, Overview, Edit Treatment Treatment Information region</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing Treatment pages by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**

To make only minor user interface changes to Treatment pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the marketing treatment regions listed above, first create your custom fields by selecting the Fields node under the Marketing Treatment object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Treatments regions listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Treatments regions. You access the configuration pages in the Application Composer from the **Pages** node under the **Marketing Treatment** object.

To access the Pages Overview page:

1. Select the **Marketing** application on the main Overview page.
2. In the object tree, select the **Marketing Treatment** object.
3. Select the **Pages** node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Treatments region that you want to customize.
Customizing Marketing Response Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Response object, which you then add for display in the Response Processing work area. Marketing Response does not have any child objects. This topic describes the extensible options for marketing responses.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

**Extensibility: Quick Steps**

This section provides a brief overview of the steps to customize the response processing work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then extend the Marketing Response node.

    Note

    If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the Fields node to create custom fields, or modify standard fields.

3. Choose the Pages node to show or hide fields, either custom or standard.

4. Test your work by navigating to the Response Processing work area to view your changes.

**Understanding Which Oracle Fusion Marketing Response Pages Are Extensible**

To customize marketing Response pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing response pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these tabs, you must select the Marketing Response object in the Application Composer and create your custom fields.
<table>
<thead>
<tr>
<th>Marketing Response Configuration</th>
<th>Response Processing UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Manage Responses, Responses, Responses table details</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Manage Responses, Create Response, Summary region</td>
</tr>
<tr>
<td></td>
<td>Manage Responses, Edit Response, Summary region</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Manage Response page by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**

To make only minor user interface changes to Manage Response pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the Marketing Response regions listed above, first create your custom fields by selecting the Fields node under the marketing response object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Manage Response region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Manage Response regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Response object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Response object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Manage Response region that you want to customize.

**Customizing Marketing Budget Pages Using Application Composer: Explained**

Use the Application Composer to create custom fields for the Marketing Budget object, which you then add for display in the Budgeting work area. This topic describes the extensible options for marketing budgets.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.
Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Campaigns work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Budget Version 2 node.

   **Note**
   If you can't see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the Fields node to create custom fields, or modify standard fields.
3. Choose the Pages node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Budgeting work area to view your changes.

Understanding Which Oracle Fusion Marketing Budget Pages Are Extensible

To customize marketing budget pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing budget pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these pages, you must select the Marketing Budget object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Budget Configuration</th>
<th>Marketing Budget UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Budgets, Overview summary table</td>
</tr>
<tr>
<td>Edit Creation Page</td>
<td>Budgets, Create Budget creation page</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Budgets, Edit Budget page</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Manage Budgets page by accessing the appropriate Application Composer configuration page, listed in the following section.

**Note**
To make only minor user interface changes to Marketing Budget pages without creating new objects or fields, use Page Composer instead of the Application Composer.
**Adding or Modifying Fields**

To add custom fields to the marketing budget regions listed above, first create your custom fields by selecting the Fields node under the Marketing Budget object. Then, select either a standard field to modify, or create a custom field.

**Adding Your Changes to the Runtime Application**

To add custom fields to the Budget region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Budget regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Budget object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Budget Version 2 object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Budget region that you want to customize.

**Customizing Marketing Budget Entry Pages Using Application Composer: Explained**

Use the Application Composer to create custom fields for the Marketing Budget Entry object, which you then add for display in the Budgeting work area. This topic describes the extensible options for marketing budget entries.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

**Extensibility: Quick Steps**

This section provides a brief overview of the steps to customize the Campaigns work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Budget Entry Version 2 node.

**Note**

If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the Fields node to create custom fields, or modify standard fields.
3. Choose the Pages node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Budgeting work area to view your changes.

Understanding Which Oracle Fusion Marketing Budget Entry Pages Are Extensible

To customize marketing budget entry pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing budget entry pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these pages, you must select the Marketing Budget Entry object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Budget Entry Configuration</th>
<th>Marketing Budget Entry UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Budgets, Manage Budget Entries summary table</td>
</tr>
<tr>
<td>Edit Creation Page</td>
<td>Create Budget Entry creation page</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Edit Budget Entry page</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Manage Budgets page by accessing the appropriate Application Composer configuration page, listed in the following section.

Note

To make only minor user interface changes to Marketing Budget Entry pages without creating new objects or fields, use Page Composer instead of the Application Composer.

Adding or Modifying Fields

To add custom fields to the marketing budget entry regions listed above, first create your custom fields by selecting the Fields node under the Marketing Budget Entry object. Then, select either a standard field to modify, or create a custom field.

Adding Your Changes to the Runtime Application

To add custom fields to the budget entry region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired budget entry regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Budget Entry Version 2 object.

To access the Pages Overview page:
1. Select the **Marketing** application on the main Overview page.

2. In the object tree, select the **Marketing Budget Entry Version 2** object.

3. Select the **Pages** node.

4. On the Pages Overview page, select the configuration page hyperlink related to the Budget Entry region that you want to customize.

### Customizing Marketing Claim Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Claim object, which you then add for display in the Budgeting work area. This topic describes the extensible options for marketing claims.

To access the Application Composer, select **Application Composer** from the **Navigator** menu, under the **Tools** category.

#### Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Marketing Claim work area.

Follow these steps:

1. Navigate to the **Application Composer**, select **Marketing** as the Application, expand the **Standard Objects** tree node, then expand the **Marketing Claim Version 2** node.

#### Note

If you can't see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the **Fields** node to create custom fields, or modify standard fields.

3. Choose the **Pages** node to show or hide fields, either custom or standard.

4. Test your work by navigating to the marketing claim work area to view your changes.

#### Understanding Which Oracle Fusion Marketing Claim Pages Are Extensible

To customize marketing claim pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record
The following table lists marketing claim pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these pages, you must select the Marketing Claim object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Claim Configuration</th>
<th>Marketing Claim UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Budgets, Manage Claims summary table</td>
</tr>
<tr>
<td>Edit Creation page</td>
<td>Create Claim creation page</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Budgets, Edit Claim page</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing Claims page by accessing the appropriate Application Composer configuration page, listed in the following section.

Note
To make only minor user interface changes to Marketing Claim pages without creating new objects or fields, use Page Composer instead of the Application Composer.

Adding or Modifying Fields
To add custom fields to the marketing claim pages listed above, first create your custom fields by selecting the Fields node under the Marketing Claim object. Then, select either a standard field to modify, or create a custom field.

Adding Your Changes to the Runtime Application
To add custom fields to the Marketing Claim page listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired marketing claim pages. You access the configuration pages in the Application Composer from the Pages node under the Marketing Claim object.

To access the Pages Overview page:
1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Claim Version 2 object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the marketing claim region that you want to customize.

Customizing Marketing Claim Settlement Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Claim Settlement object, which you then add for display in the Budgeting work area. This topic describes the extensible options for marketing claim settlements.
To access the Application Composer, select **Application Composer** from the **Navigator** menu, under the **Tools** category.

**Extensibility: Quick Steps**

This section provides a brief overview of the steps to customize the Marketing Claim Settlement work area.

Follow these steps:

1. Navigate to the **Application Composer**, select **Marketing** as the Application, expand the **Standard Objects** tree node, then expand the **Marketing Claim Settlement Version 2** node.

   **Note**

   If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the **Fields** node to create custom fields, or modify standard fields.

3. Choose the **Pages** node to show or hide fields, either custom or standard.

4. Test your work by navigating to the marketing claim settlement work area to view your changes.

**Understanding Which Oracle Fusion Marketing Claim Settlement Pages Are Extensible**

To customize marketing claim settlement pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing claim settlement pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these pages, you must select the Marketing Claim Settlement object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Claim Settlement Configuration</th>
<th>Marketing Claim Settlement UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Creation page</td>
<td><strong>Create Claim</strong> creation page</td>
</tr>
<tr>
<td><strong>Edit Summary Form</strong> details page</td>
<td><strong>Budgets, Edit Claim, Settlement</strong> tab</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Marketing Claim Settlement page by accessing the appropriate Application Composer configuration page, listed in the following section.

**Note**
To make only minor user interface changes to Marketing Claim Settlement pages without creating new objects or fields, use Page Composer instead of the Application Composer.

Adding or Modifying Fields

To add custom fields to the marketing claim regions listed above, first create your custom fields by selecting the Fields node under the Marketing Claim Settlement object. Then, select either a standard field to modify, or create a custom field.

Adding Your Changes to the Runtime Application

To add custom fields to the Marketing Claim Settlement region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired claim settlement regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Claim Settlement object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Claim Settlement Version 2 object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the marketing claim settlement region that you want to customize.

Customizing Marketing Fund Request Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the Marketing Fund Request object, which you then add for display in the Budgeting work area. This topic describes the extensible options for marketing fund requests.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the Fund Request work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Marketing Fund Request Version 2 node.
If you can't see the choice list of applications, make sure the regional area of the page is exposed.

2. Choose the **Fields** node to create custom fields, or modify standard fields.
3. Choose the **Pages** node to show or hide fields, either custom or standard.
4. Test your work by navigating to the marketing fund request work area to view your changes.

**Understanding Which Oracle Fusion Marketing Fund Request Pages Are Extensible**

To customize marketing fund request pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists marketing fund request pages and regions that you can customize using Application Composer. To create custom fields that you can later add to any of these pages, you must select the Marketing Fund Request object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Marketing Fund Request Configuration</th>
<th>Marketing Fund Request UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Budgets, Manage Fund Requests summary table</td>
</tr>
<tr>
<td>Edit Creation page</td>
<td>Create Fund Request creation page</td>
</tr>
<tr>
<td>Edit Summary Form details page</td>
<td>Budgets, Edit Fund Request page</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Fund Request page by accessing the appropriate Application Composer configuration page, listed in the following section.

**Note**

To make only minor user interface changes to Marketing Fund Request pages without creating new objects or fields, use Page Composer instead of the Application Composer.

**Adding or Modifying Fields**

To add custom fields to the marketing fund request regions listed above, first create your custom fields by selecting the Fields node under the Marketing Fund Request object. Then, select either a standard field to modify, or create a custom field.
Adding Your Changes to the Runtime Application

To add custom fields to the marketing fund request region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the desired marketing fund request regions. You access the configuration pages in the Application Composer from the Pages node under the Marketing Fund Request object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page.
2. In the object tree, select the Marketing Fund Request Version 2 object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Marketing Fund Request region that you want to customize.

Customizing Sales Lead Pages Using Application Composer: Explained

Use the Application Composer to create custom fields for the marketing lead object, which you then add for display in the Lead work area. This topic describes the extensible options for sales lead and associated objects.

To access the Application Composer, select Application Composer from the Navigator menu, under the Tools category.

The Oracle Fusion Marketing objects that are associated with the CRM Lead Management application are:

- Sales Leads
- Sales Lead Contacts (child object of Sales Lead)
- Sales Lead Resources (child object of Sales Lead)
- Marketing Response (related object of Sales Lead)

Extensibility: Quick Steps

This section provides a brief overview of the steps to customize the sales lead work area.

Follow these steps:

1. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand Sales Lead, Sales Lead Contacts, or Sales Lead Resources nodes.

Note

If you can't see the choice list of applications, make sure the regional area of the page is exposed.
2. Choose the **Fields** node to create custom fields, or modify standard fields.
3. Choose the **Pages** node to show or hide fields, either custom or standard.
4. Test your work by navigating to the Sales Lead work area to view your changes.

**Understanding Which Oracle Fusion Sales Leads Pages Are Extensible**

To customize Sales Leads pages, you need to know which pages and regions are extensible, and which objects to select in the Application Composer to customize those pages. Pages represent the ways in which object information can be displayed.

For example, use:

- Landing pages to show lists of object records
- Creation pages to create new object records
- Detail pages show additional details about an individual object record

The following table lists Sales Lead pages, regions, child objects and the related object that you can customize using Application Composer. For example, the Sales Lead Contacts tab on the Edit Sales Lead page is a child object of the Sales Lead object and is extensible. To create custom fields that you can later add to the Sales Lead Contacts tab, you must select the Sales Lead Contacts object in the Application Composer and create your custom fields.

<table>
<thead>
<tr>
<th>Sales Leads Configuration</th>
<th>Sales Leads UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Summary Table overview page</td>
<td>Lead Qualification, Lead table</td>
</tr>
<tr>
<td>Create Lead creation page</td>
<td>Lead Qualification, Create Lead page</td>
</tr>
<tr>
<td>Registration Details creation page</td>
<td>Lead Qualification, Create Lead, Registration Details region (select Partner as the Sales Channel)</td>
</tr>
<tr>
<td>Show Details details page</td>
<td>Lead Qualification, Edit Lead, Summary region (expand Show Details to view more details)</td>
</tr>
<tr>
<td>Basic Qualification details page</td>
<td>Lead Qualification, Edit Lead (expand the Basic Qualification option to view details)</td>
</tr>
<tr>
<td>Registration Details details page</td>
<td>Lead Qualification, Edit Lead (when Sales Channel is Partner). Expand the Registration Details option to view details.</td>
</tr>
<tr>
<td>Edit Lead details page</td>
<td>Lead Qualification, Edit Lead, Summary region</td>
</tr>
<tr>
<td>Contacts lead tab component</td>
<td>Lead Qualification, Edit Lead, Contacts tab</td>
</tr>
<tr>
<td>Resources lead tab component</td>
<td>Lead Qualification, Edit Lead, Sales Team tab</td>
</tr>
<tr>
<td>Marketing Response lead tab component</td>
<td>Lead Qualification, Edit Lead, Responses tab</td>
</tr>
</tbody>
</table>

Next, expose those custom fields on the Sales Lead page by accessing the appropriate Application Composer configuration page listed in the following section.

**Note**

To make only minor user interface changes to Sales Leads pages without creating new objects or fields, use Page Composer instead of the Application Composer.
Adding or Modifying Fields

To add custom fields to the Sales Lead regions listed above, first create your custom fields by selecting the Fields node under the sales lead, or sales contact, or sales resources object. Then, select either a standard field to modify, or create a custom field.

Adding Your Changes to the Runtime Application

To add custom fields to the Sales Lead Contacts region listed in the table above, first create your custom fields using the Application Composer. Next, use the Application Composer’s configuration pages to add those custom fields to the desired Sales Leads regions. You access the configuration pages in the Application Composer from the Pages node under the Sales Lead object.

To access the Pages Overview page:

1. Select the Marketing application on the main Overview page
2. In the object tree, select the Sales Lead object.
3. Select the Pages node.
4. On the Pages Overview page, select the configuration page hyperlink related to the Sales Lead region that you want to customize.

Creating a Sales Lead Validation Rule Using Application Composer: Worked Example

Lead management users can change the status of a lead to Qualified, regardless of whether the lead customer is a sales account or has a primary product associated with the lead. However, to enforce compliance with your company’s lead management business processes, you might want to create business rules to control when a lead’s status can actually be changed to Qualified.

To do this, use the Application Composer to create validation rules to enforce certain criteria before allowing a sales lead status to be changed to Qualified.

In this example, you will use the Application Composer to create a validation rule that ensures a primary product and sell-to address exist, before a lead’s status can be changed to Qualified.

Create a validation rule for Sell-to Address

1. Go to Navigator, Application Composer.
2. From the Application field, select Marketing.
3. From the Objects View, select Standard Objects, Sales Lead, Server Scripts.
4. Click the Object Functions tab, and then click the Add a New Object Function icon to go to the Create Object Function screen.
5. In the Function Name field, enter the following name without spaces: isSellToExists In the Returns field, select Boolean.
6. In the Function Body area enter the following:
   ```java
   if(null != CustomerId){ def partySites = newView('Address'); def
   ```
Define Extensions for Marketing

```java
criteria = partySites.createViewCriteria(); def criteriaRow = criteria.createRow(); criteria.insertRow(criteriaRow); def criteriaItem = criteriaRow.ensureCriteriaItem('PartyId'); criteriaItem.setValue(CustomerId)
partySites.appendViewCriteria(criteria) partySites.executeQuery();
while(partySites?.hasNext()) { def partySite = partySites.next();
def partySiteUses = partySite?.getAttribute('PartySiteUse');
while (partySiteUses?.hasNext()) { def partySiteUse = partySiteUses.next(); if ('SELL_TO' ==
partySiteUse?.getAttribute('SiteUseType')) { return true; } } }
return false;
```

7. Click Validate.
8. Click Save and Close.

Create a rule for enforcing the Primary Product association for the sales lead

Next, you want to create a rule for enforcing the primary product association for the sales lead as follows:

1. Click the Add a New Object Function icon to go to the Create Object Function screen.
2. In the Function Name field, enter the following name without spaces: isPrimaryProductAssigned. In the Returns field, select Boolean.
3. In the Function Body area enter the following: if((null !=
PrimaryInventoryOrgId && null != PrimaryInventoryItemId) || null !=
PrimaryProductGroupId) return true; else return false;
4. Click Validate.
5. Click Save and Close to return to the Server Scripts Sales Lead screen.

Create a validation rule to check for Lead qualification

1. Click the Validation Rules tab and from the Object Rules area, click the Add a new validation rule icon.
2. In the Create Object Validation Rule screen, go to the Rule Name field and enter the following rule name without spaces: qualifyLead
3. In the rule definition area, enter the following:
   ```java
   if(isAttributeChanged('StatusCode') && getAttribute('StatusCode')
   == 'QUALIFIED') return (isPrimaryProductAssigned() &&
isSellToExists()); else return true;
   ```
4. Go to the Error Message section and enter the following message text: Primary Product and Sell to Address are required for qualifying a Lead.
5. Click Save and Close to complete the task of using the Application Composer to create validation rules for the primary product and sell-to address fields when setting the lead to qualified.

Test the rule

1. Go to Navigator, Lead Qualification, Edit Lead page.
2. From the Actions menu, select Qualify.
If the Primary Product and Sell to Address fields contain no data, you should receive the following error message text:

Primary Product and Sell to Address are required for qualifying a Lead.

Adding a Field to a Sales Lead Page: Worked Example

This topic explains how to add a custom number field to an existing user interface page in the Lead Management application.

You want to add a field to the Header Details section of the Sales Lead UI. You have a requirement to enter a reference number to associate to the date that you manually created a lead in the format of day, month, year. For example, a lead that you manually created on the first day of April 2000 would be referenced by the number 01042000.

Add a Custom Number Field to a Sales Lead Page

To add a custom number field to a Sales Lead overview page, perform the following steps:

1. Select Application Composer from the Navigator menu, under the Tools category.
2. Next select Administration, Manage Sandboxes... and click New to create a new sandbox.
3. Enter the sandbox name and a short description, then click Save and Close.
4. Click OK when the confirmation dialog displays.
5. From the list of available Sandboxes, highlight the sandbox name that you created and click Set as Active.
6. Navigate to the Application Composer, select Marketing as the Application, expand the Standard Objects tree node, then expand the Sales Lead node.

If you can’t see the choice list of applications, make sure the regional area of the page is exposed.

7. Click Fields and then from the Fields screen, select Create to create a custom field.
8. From the Select Field Type dialog, select the Number type of field and click OK.
9. In the Display Label field of the Appearance area, enter the label name that you want displayed on the header details of the Sales Lead user interface. For example, enter Lead Reference.

The Name field in the Name area is automatically populated with the value you entered in the Display Label field. However, this field requires a unique name in the system. Name and description are for internal use only, and are not displayed in the user interface.
10. In the **Constraints** area, select Required since you want to ensure the user enters a reference number when creating a lead.

11. Enter the maximum and minimum values for the number of digits you can enter for the number.

12. Click **Save and Close**.

13. Log out and log back in again to the sandbox environment where you created the lead reference number field.

   Once you are satisfied with the customizations made in the sandbox, you can replicate your changes in the sandbox, and then publish them to the mainline.

14. Navigate to the Sales Lead work area to view and validate your changes.

**FAQs for Define Extensions for Marketing**

**What's the difference between Page Composer and Application Composer?**

Page Composer is an Oracle Fusion tool you can use to modify Oracle Fusion user interface (UI) pages and components for all products designated for use with Page Composer. Page Composer uses two different modes of Design View. The first mode, Design View: Standard mode, is selected by default in all CRM pages when opening a page with Page Composer with the Design button selected. The second mode, Design View: Direct Selection mode, is activated when you click the Select tab for the UI page you want to customize. In CRM, Direct Selection mode is available when you customize pages, but not when you personalize a dashboard page. With the Design View: Direct Selection mode, you can select and edit UI elements such as form fields and table columns. In Direct Selection mode, selectable UI components become apparent when you move your cursor over the UI component. Selectable UI components are highlighted and can be edited.

The following table describes how you can use each mode of Page Composer to customize dashboard pages and other select pages (such as the Partner Public Profile page, Partner Landing page, Partner Registration, Customer Snapshot, and Customer Overview - Analysis tab), and customize transactional pages (all other non-dashboard pages).

<table>
<thead>
<tr>
<th>Use Cases</th>
<th>Design View - Standard mode</th>
<th>Design View - Direct Selection mode</th>
<th>Page Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add content (Business Intelligence reports, CRM portlets such as Calendar)</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Delete region</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Move region</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
</tbody>
</table>
The Oracle Fusion CRM Application Composer also lets you make UI changes at run time. However, the types of UI changes that you can make using the Application Composer are quite different. Specifically, your primary focus when using the 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 Application Composer, you cannot access the Resource Catalog to add new content to a page. With the Application Composer, administrators can make customizations at the site level only.

The following table describes some of the primary differences between Page Composer and the Application Composer.

<table>
<thead>
<tr>
<th>Customization Task</th>
<th>Available in Page Composer (site, job role, external or internal level)?</th>
<th>Available in Application Composer (site level only)?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change page layout (for example, change a two column layout to three column layout)</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Default region state (open or close)</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Manage save queries (create and edit)</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Hide or show field</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Change field label</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Make field required or not</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Make field read-only or updateable</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Reorder fields in a Form</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Reorder table columns</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Hide or show table columns</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Set table column width with the mouse</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Set table column width and min width in percent or pixels</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Make column sortable or not</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<p>| The Oracle Fusion CRM Application Composer also lets you make UI changes at run time. However, the types of UI changes that you can make using the Application Composer are quite different. Specifically, your primary focus when using the 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 Application Composer, you cannot access the Resource Catalog to add new content to a page. With the Application Composer, administrators can make customizations at the site level only. |</p>
<table>
<thead>
<tr>
<th>Feature</th>
<th>Option 1</th>
<th>Option 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reorder subtabs</td>
<td>No</td>
<td>Yes</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>Simple field customizations (show, hide, make read only, make required)</td>
<td>Yes (WYSIWYG - what you see is what you get)</td>
<td>Yes (non-WYSIWYG)</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 Page Composer design interface</td>
<td>Yes, in the CRM application that you are customizing</td>
</tr>
</tbody>
</table>
Use web services to integrate web-based applications into your Oracle Fusion applications. Web services expose Oracle Fusion Applications business objects and processes to other applications through the use of open standards-based technologies. Some of these technologies include Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Business Process Execution Language (BPEL), Web Services Description Language (WSDL), and XML schema definitions (XSD). Oracle Fusion Applications web services support development environments and clients that comply with these open standards.

Oracle Fusion Applications includes two types of web services: Application Development Framework (ADF) services and composite services. The following table describes the two types.

<table>
<thead>
<tr>
<th>Web Service Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADF services</td>
<td>ADF services usually represent business objects, such as employees or purchase orders. ADF services typically expose standard operations, such as create, update, and delete. However, for locally-persisted objects, ADF services are not limited to these operations. &lt;br&gt;Examples of ADF services include: &lt;br&gt;• Worker.changeHireDate - a service that updates the hire date of the worker business object. &lt;br&gt;• ProjectTask.createTask - a service that adds a task to the project task business object.</td>
</tr>
</tbody>
</table>
Composite services usually represent end-to-end business process flows that act on business events produced by the ADF services. Composite services orchestrate multiple object-based services, rules services, and human workflows. Examples of composite services include:

- `ProjectStatusChangeApproval.process` - a service that accepts the change in project status.
- `ScheduleOrchestrationOrderFulfillmentLineService.scheduleOrders` - a service that schedules resources used to fulfill an order.

Access Oracle Enterprise Repository for Oracle Fusion Applications to find detailed information about integration assets, such as web services. To view lists of web services, select these asset types:

- ADF Service
- ADF Service Data Object
- Composite Service
- Composite

Service methods and parameters, the service path, the WSDL URL and other technical data, appear on the Detail tab of each web service. Step-by-step instructions regarding the invocation of a service and the service XSD appear on the Documentation tab.

**Files for Import and Export**

**Files for Import and Export: Highlights**

The File Import and Export page accesses repositories of content. For example, each Oracle Fusion Applications instance connects to a single Oracle WebCenter Content server for content management.

The following documents describe use and administration of content management:

- Oracle WebCenter Content User's Guide for Content Server
- Oracle WebCenter Content System Administrator's Guide for Content Server

**Using Content Management**

- For information about what objects to upload and download, including templates for external data integration, refer to the Oracle Enterprise Repository for Oracle Fusion Applications.
For general access to content management, including to all metadata and to manage accounts, use the Oracle WebCenter Content Server’s standard service user interface.

See: Oracle WebCenter Content User’s Guide for Content Server

For information on creating accounts in WebCenter Content accounts, refer to WebCenter Content System Administrator’s Guide for Content Server.

See: Accounts

For information about naming accounts involved with import and export, see Files for Import and Export: Points to Consider.

For programmatic upload and download to content management, refer to Oracle WebCenter Content System Administrator’s Guide for Content Server.

See: About Batch Loading

Security in Content Management

For information about security, see the Security tab in Oracle Enterprise Repository for Oracle Fusion Applications.

For information about roles such as the integration specialist roles for each product family, see the Oracle Fusion Applications security reference manuals for each offering. For example:


Files for Import and Export: Explained

You can import data into or export data out of Oracle Fusion Applications using repositories of content and processes for import and export.

Integration specialists stage data for import and export. Application administrators run processes to import data in repositories of content to application transaction tables, or retrieve data exported from applications.

Aspects of managing files for import and export involve the following.

- The File Import and Export page
- Interacting with content management
- Uploading for import
- Downloading for export
- File size

The File Import and Export Page

The File Import and Export page lets you upload content to, or download content from the document repository of Oracle WebCenter Content Management.
Search criteria on the page are limited to the minimum metadata of content management records needed for file import and export.


Interacting with Content Management

Everyone who uses the File Import and Export page is assigned to one or more accounts in content management.

Accounts organize and secure access to content items.

Uploading for Import

Uploading a file creates a record.

When you create a record, you must specify an account as well as the file. When you create a record, you must specify an account as well as the file. The account you specify determines which import process picks up that file to import it.

You can upload any file formats that can be parsed by the content repository being used, such as any MIME or content types. However, the format uploaded should conform to the requirements of the import process being used, such as a comma-separated values (CSV) file for the Load Interface File for Import process.

Downloading for Export

Processes you run to export data result in files in content management. Records in the search results table of the File Import and Export page provide links to the files for download.

Note

The owner of a data export file can be an application ID (APPID).

File Size

Upload and download does not intentionally apply the following:

- Data compression
- File chunking or splitting

The `UPLOAD_MAX_DISK_SPACE` parameter in the `web.xml` file determines the maximum allowable file size in content management. The default maximum size is 10240000 (10MB).

Files for Import and Export: Points to Consider

Interaction between the File Import and Export page and Oracle WebCenter Content requires securing content in an account. Oracle provides predefined accounts in Oracle WebCenter Content.

Areas of file import and export to consider involve the following.
Security

The duty role needed for accessing the File Import and Export page is File Import and Export Management duty. This duty role is included in the predefined role hierarchy for integration specialist roles and product family administrator roles.

Files in Oracle WebCenter Content are associated with an account so that only users who have permission to a particular account can work with content items that belong to that account. You can only upload and download files to and from content management that are associated with accounts that you are entitled to access.

Oracle WebCenter Content does not support trailing slashes (/). Account names are appended with a $ to ensure each account is unique. Account names are dynamic so that if they overlap (one name is completely contained in another, longer name, such as US and USSales), each account is treated as discrete by access grants.

Security such as virus scanning is handled by the underlying integrated content management.

Searching Records

A record in Oracle WebCenter Content contains metadata used for accessing the file.

When a scheduled process has run to completion on a file, the record for the file includes a process ID.

Accessing Content in a New Account

When you create a new account in Oracle WebCenter Content and the content server is not restarted, access to content in the new account from the File Import and Export page may be delayed until the policy store is updated.

Account Names

If you create custom accounts for importing or exporting data, use the following conventions for naming the account: Do not include a slash "/" at the beginning or end. End with "$" to avoid partial string matching. Use "$/" as a separator in the hierarchical structure.

For example: fin$/journal$/import$. The File Import and Export page transforms account names by removing the $s. For example fin$/journal$/import$ displays as fin/journal/import. The Remote Introduc Client (RIDC) HTTP command-line interface (CLI) transforms the account name you specify without $ symbols to one that includes them. For example, fin/journal/import becomes fin$/journal $/import$ in WebCenter Content.
Deleting Files
You can delete one file at a time when you use the File Import and Export page. To delete multiple files simultaneously from the content repository, use the standard service page in Oracle WebCenter Content.

External Data Integration Services for Oracle Cloud

External Data Integration Services for Oracle Cloud: Overview

Use External Data Integration Services for Oracle Cloud to load data into Oracle Fusion Applications from external sources, such as legacy systems and third-party applications.

Components of External Data Integration Services for Oracle Cloud include:

- Templates and control files for formatting, structuring, and generating the data file.
- A general file load process for loading values from the data file into interface tables.
- Application-specific data import processes for transferring data from interface tables to the application tables in your Oracle Fusion Applications.

To use External Data Integration Services for Oracle Cloud to load data into Oracle Fusion Applications tables:

1. Prepare your data and generate a data file by using the product-specific templates and control files.
2. Transfer the data file to the integrated content management server.
3. Run the Load Interface File for Import process.
4. Correct data load errors, if necessary.
5. Run the appropriate application-specific process for validating and inserting the data into application tables.
6. Correct data import errors, if necessary.

For templates and control files, see assets with the File-Based Data Import type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com). For more information, see the Documentation tab for the Load Interface File for Import process in Oracle Enterprise Repository.

Locating File Import Templates: Explained

External data that you integrate into your Oracle Fusion Applications must be structured and formatted according to the properties of the fields and tables that store the data. To prepare external data so that data types, structural
relationships, and other properties of the data correctly align to the data types, structural relationships, and properties of the target tables, use the product-specific templates and control files in Oracle Enterprise Repository for Oracle Fusion Applications.

You access these files from the Documentation tab of the scheduled process that corresponds to the interface tables that store the data. To find the process, you can search the interface table or you can search the specific process, if you know it.

Aspects of preparing external data using templates involve these tasks.

- Finding templates and control files
- Downloading templates
- Opening XLS templates

**Finding Templates and Control Files**

To find the templates and control files:

1. Sign in to Oracle Enterprise Repository.
2. Enter the following information in the Search fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search String</td>
<td>FBDI</td>
</tr>
<tr>
<td>Type</td>
<td>Scheduled Process</td>
</tr>
<tr>
<td>FusionApps: Logical Business Area</td>
<td>(Optional) Select the value relevant to your implementation.</td>
</tr>
</tbody>
</table>

3. Click Search.
4. Select Load Interface File for Import from the results.

**Downloading Templates**

To download the templates:

1. Use the Search area to locate the Load Interface File for Import job and then select it from the search results.
2. Click the Documentation tab in the lower pane to see a list of links to application-specific import jobs.
3. Click a link to access the job.
4. Click the Documentation tab in the lower pane to see a list of links that access:
   - Control files, which describe the logical flow of the data load process
   - XLS templates, which include worksheets and macros that assist you in structuring, formatting, and generating your data file
5. Click the link to download the file.

**Opening the XLS Template**

To prepare your data in a spreadsheet format, use XLS templates:
1. Open the XLS template.
   The first worksheet in each file provides instructions for using the template.

   **Important**
   If you omit or fail to complete the instructions, data load errors and data import failure are likely.

2. Save a copy of the file.
3. Click the **Generate CSV File** button.
   The macro generates a comma-separated values (CSV) file and compresses it into a ZIP file; you must transfer the ZIP file to the content management server.

---

**Using Excel Integration Templates to Generate Data Files: Points to Consider**

Oracle Enterprise Repository for Oracle Fusion Applications includes integration templates to help you prepare external data for loading and importing. Each template includes table-specific instructions, guidelines, formatted spreadsheets, and best practices for preparing the data file for upload. Use the templates to ensure that your data conforms to the structure and format of the target application tables.

**Templates**

This list details the characteristics of the templates:

- Each interface table is represented by a separate worksheet.
- Each interface table field is represented by a worksheet column with a header in the first row.
- Each column header contains bubble text, or comments, that include details about the column, such as the expected data type, length, and, in some cases, other instructional text.
- The worksheet columns appear in the order that the control file processes the data file.
- The columns that you do not intend to use can be hidden, but not reordered or deleted.

   **Important**
   Deleting or reordering columns will cause the load process to fail and result in an unsuccessful data load.

- The external data must conform to the data type that the control file and process for the associated database column accepts.
- Date column values must appear in the YYYY/MM/DD format.
• Amount column values must appear with no separators other than a period (.) as the decimal separator.
• Negative values must be preceded by the minus (-) sign.
• Column values that require whole numbers include data validation to allow whole numbers only.
• Columns are formatted, where applicable, to match the target field data type to eliminate data entry errors.
• For columns that require internal ID values, refer to the bubble text for additional guidance about finding these values.
• When using Microsoft Excel to generate or update the CSV file, you must select YYYY/MM/DD as your regional setting for date values.

Using XML Templates to Generate Data Files for Integration: Highlights

Oracle Enterprise Repository for Oracle Fusion Applications includes XML integration templates assets that you use with Oracle Data Integrator (ODI) to generate import files from your external data.

To use the XML templates and generate the import files, you must:

• Install and set up Oracle Data Integrator
• Create source and target models
• Create integration projects

Note

In Oracle Cloud implementations, you must upload the ZIP file to the content management repository in Oracle Cloud. In non-Cloud implementations, you can streamline the data integration process by installing the content management document transfer utility so ODI performs the ZIP file transfer.

Oracle Data Integrator provides a solution for integrating complex data from a variety of sources into your Oracle Fusion applications. The Oracle Fusion Middleware Installation Guide for Oracle Data Integrator and the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator provide complete details pertaining to the installation and set up of this product.

Installing and Setting Up Oracle Data Integrator

• Install Oracle Data Integrator to use Oracle Fusion Applications XML integration templates. Refer to the Oracle Fusion Middleware Installation Guide for Oracle Data Integrator.
  See: Installing Oracle Data Integrator
• Set up Oracle Data Integrator to use Oracle Fusion Applications XML integration templates. Refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.
  See: Setting up the Topology
Creating Source and Target Models

- Create the ODI models for both the source and target datastores. You determine the source models that you use based on the system or technology of the external data that you import into your Oracle Fusion application. You create the target models by importing the XML files, which you download from Oracle Enterprise Repository. For more information, refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.

See: Creating and Reverse-Engineering a Model

Configuring Integration Projects

- Create and configure an integration project, which entails selecting the knowledge modules, creating the interfaces, and mapping the source and target datastores. For more information, refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.

See: Creating an Integration Project

Using XML Integration Templates to Generate Data Files: Points to Consider

Use XML templates in Oracle Data Integrator to prepare your external data for loading and importing. Oracle Enterprise Repository for Oracle Fusion Applications includes three types of XML templates that you import as target models in your Oracle Data Integrator repository.

Oracle Enterprise Repository includes these three levels of XML files:

- Family-level
- Product-level
- Product

Family-Level XML Files

A family-level XML file is common to a group of product-level model folders and product models.

Consider the following points when you use family-level XML files:

- The family-level XML file supports all of the Oracle Enterprise Repository assets in the family, for example Oracle Fusion Financials or Human Capital Management.
- You import the family-level XML file into your Oracle Data Integrator repository prior to importing the other XML files.
- You import one family-level XML file as a model folder for each family of products.
- You import each family-level XML file as a top-level model folder.
- You import the family-level XML file one time; it supports all subsumed product-level model folders.
• You select Synonym mode Insert Update as the import type.

Product-Level XML Files

A product-level XML file is common to a group of product models.

Consider the following points when you use product-level XML files:

• The product-level XML file supports all of the Oracle Enterprise Repository assets in the product line, for example Fixed Assets, General Ledger, or Payables.

• You import one product-level XML file as a model folder for each line of products.

• You import the product-level XML file as a model folder into your Oracle Data Integrator repository after you import the family-level XML file, but before you import product XML files.

• You import each product-level XML file as a midlevel model folder within the appropriate family-level model folder.

• You import the product-level XML file one time; it supports all subsumed product models.

• You select Synonym mode Insert Update as the import type.

Product XML Files

A product XML file represents a specific Oracle Enterprise Repository interface table asset.

Consider the following points when you use product XML files:

• You import one product XML file as a model for each interface table or set of tables, for example Mass Additions.

• You import the product XML file as a model into your Oracle Data Integrator repository after you import the product-level XML file.

• You import each product XML file as a model within the appropriate product-level model folder.

• You import each product XML file one time.

• You select Synonym mode Insert Update as the import type.

• The model is based on File technology.

• After you import the product model, you connect the model to the correct logical schema.

Creating Integration Projects That Generate Data Files for Import: Points to Consider

When you use Oracle Data Integrator (ODI) to generate import data files from external data sources, you must configure an integration project. Integration
projects are collections of ODI components that provide the procedural details of an integration from a source to a target. In this scenario, the source is your external data and the target is the import data file that you load and import into your Oracle Fusion Applications.

Configure these components to create your integration project.

- Knowledge modules
- Integration interfaces

**Knowledge Modules**

Knowledge modules contain the information that Oracle Data Integrator requires to perform a specific set of tasks against a specific technology or set of technologies. Examples include Check knowledge modules, which ensure that constraints on the sources and targets are not violated and integration knowledge modules, which load data to target tables.

Consider the following points about knowledge modules:

- The knowledge modules that you import into your integration project depend on the source and target technologies, as well as other integration-specific rules and processes.
- Multiple types of knowledge modules exist in ODI.
- Use the SQL File to Append module to create the import data file.

**Integration Interfaces**

Integration interfaces comprise sets of rules that define the loading of data from one or more sources to the target.

Consider the following points about integration interfaces:

- The source is the datastore from your external data model.
- The target is the interface table datastore, which is the CSV file from your interface table model.
- After you set up the source and target datastores, map the target fields to the source fields.
- You can map source field values to target fields or constants.

**Transferring Data Files to Target Accounts in Oracle WebCenter Content: Explained**

After you generate the ZIP file that contains the CSV data import file, transfer it to the content repository.

Use any of these methods to transfer file:

- File Import and Export page in Oracle Fusion Applications
- Oracle WebCenter Content Document Transfer Utility
• Oracle Fusion Financials Utility web service

**Note**

Consult Oracle Enterprise Repository for Oracle Fusion Applications for web service documentation.

Aspects of transferring data files to content management involve the following:

• Target accounts

• Accessing transferred content

**Target Accounts**

You must transfer files to these predefined account in content management that corresponds to the interface table or assets.

<table>
<thead>
<tr>
<th>Interface Table</th>
<th>Predefined Account</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payables Standard Invoice Import</td>
<td>fin/payables/import</td>
</tr>
<tr>
<td>• AutoInvoice Import</td>
<td>fin/receivables/import</td>
</tr>
<tr>
<td>• Receivables Standard Receipt Import</td>
<td></td>
</tr>
<tr>
<td>• Customer Import</td>
<td></td>
</tr>
<tr>
<td>• China Value Added Tax Invoice Import</td>
<td></td>
</tr>
<tr>
<td>• BAI2 Format Bank Statements Import</td>
<td>fin/cashManagement/import</td>
</tr>
<tr>
<td>• EDIFACT FINSTA Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>• ISO200022 CAMT053 Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>• SWIFT MT940 Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>• Fixed Asset Mass Additions Import</td>
<td>fin/assets/import</td>
</tr>
<tr>
<td>• Fixed Asset Mass Adjustments Import</td>
<td></td>
</tr>
<tr>
<td>• Fixed Asset Mass Retirements Import</td>
<td></td>
</tr>
<tr>
<td>• Fixed Asset Mass Transfers Import</td>
<td></td>
</tr>
<tr>
<td>• Fixed Asset Units of Production Import</td>
<td></td>
</tr>
<tr>
<td>Intercompany Transaction Import</td>
<td>fin/intercompany/import</td>
</tr>
<tr>
<td>• Journal Import</td>
<td>fin/generalLedger/import</td>
</tr>
<tr>
<td>• Chart Of Accounts Segment Values and Hierarchies Import</td>
<td></td>
</tr>
<tr>
<td>General Ledger Budget Balance Import</td>
<td>fin/budgetBalance/import</td>
</tr>
<tr>
<td>Supplier Bank Account Import</td>
<td>fin/payables/import</td>
</tr>
<tr>
<td>Tax Configuration Content Import</td>
<td>fin/tax/import</td>
</tr>
<tr>
<td>Import Blanket Purchase Agreements</td>
<td>prc/blanketPurchaseAgreement/import</td>
</tr>
<tr>
<td>Import Contract Purchase Agreements</td>
<td>prc/contractPurchaseAgreement/import</td>
</tr>
</tbody>
</table>
You can create subaccounts to further organize your files. However, you must create the account subordinate to the predefined account for the asset you are integrating.

**Accessing Transferred Content**

To access your transferred data you must access the account that corresponds to the interface table or asset appropriate for the data.

Available data integration processes move the content into and out of Oracle Fusion Applications tables. Running an import or export process creates a process ID in content management that you can use to identify the content you wish to overwrite or extract.

Oracle Enterprise Scheduler import process jobs result in the following hierarchy of items in Oracle WebCenter Content:

- A root import job is a list of all unprocessed files in an account. This job submits the child jobs that process each unprocessed file.
- A parent import job is a single file ID, account name, and the import steps (download, extract, import) for a single job, job set, or subrequests. This type of job tags the file with its request ID, provided the file is not deleted immediately after successful import.
- A child import job is a direct data load from a prepared file, typically a SQLLoader. Typically, the parent import job submits this job.

**Document Transfer Utility: Explained**

The WebCenter Content Document Transfer Utility for Oracle Fusion Applications is a feature-set Java library that provides programmatic access to
the content repository. Use the utility to import and export documents, such as import files that contain external data that you want to load into interface and application tables.

The library includes:

- Oracle WebCenter Content client command line tool
- Oracle Data Integrator (ODI) upload and download tools
- Oracle WebCenter Content remote intradoc client (RIDC)
- Oracle HTTPClient
- Oracle Fusion Applications branding and defaults

Options for the WebCenter Content Document Transfer Utility for Oracle Fusion Applications fall into these categories:

- DownloadTool program options
- UploadTool program options
- Debugging and silent invocation options

### DownloadTool Program Options

This table describes the download tool program options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>Protocol-specific connection URL of content server</td>
</tr>
<tr>
<td>username</td>
<td>Username to leverage</td>
</tr>
<tr>
<td>password</td>
<td>Password, supplied in command line</td>
</tr>
<tr>
<td>passwordFile</td>
<td>Password, supplied in text file on the first line of the file</td>
</tr>
<tr>
<td>dID</td>
<td>ID of document revision to download</td>
</tr>
<tr>
<td></td>
<td>dID is unique across repository</td>
</tr>
<tr>
<td></td>
<td>dID changes with each revision</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>Alternatively, specify the dDocName and RevisionSelectionMethod to identify the dID to leverage.</td>
</tr>
<tr>
<td>dDocName</td>
<td>Content name</td>
</tr>
<tr>
<td></td>
<td>Multiple revisions of a document can share the same dDocName value, otherwise it is unique.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>You should also provide RevisionSelectionMethod value.</td>
</tr>
</tbody>
</table>


RevisionSelectionMethod  | Revision to download
--- | ---
Valid values: Latest, LatestReleased
Default value: Latest

outputFile  | Path and name of local file to write
--- | ---

Here you see a sample download invocation command:

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic
password=we1com3i
dID=21537 outputFile="/tmp/output.doc"
```

Here you see sample output:

```
Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing download (GET_FILE) ...
Download successful.
```

**UploadTool Program Options**

This table describes the upload tool program options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>Protocol-specific connection URL of content server</td>
</tr>
<tr>
<td>username</td>
<td>Username to leverage</td>
</tr>
<tr>
<td>password</td>
<td>Password, supplied in command-line</td>
</tr>
<tr>
<td>passwordFile</td>
<td>Password, supplied in text file on the first line of the file</td>
</tr>
<tr>
<td>primaryFile</td>
<td>Fully-qualified path of local primary file to upload</td>
</tr>
<tr>
<td>dDocAccount</td>
<td>Destination account</td>
</tr>
<tr>
<td>dDocTitle</td>
<td>Document title</td>
</tr>
</tbody>
</table>
| checkout        | If uploading a document revision, check out the document from the repository before uploading the revision  
Valid values: true, false
Default value: false
| ignoreCheckoutErrorNeg22 | Ignore error -22 (user has already checked-out the document) when checking-out the document.  
Valid values: true, false
Default value: true

Here you see a sample upload invocation command:

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
oracle.ucm.client.UploadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic
password=we1com3i
```
Here you see sample output:

Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing upload (CHECKIN_UNIVERSAL) ...
Upload successful.
[dID=21537 | dDocName=UCMFA021487]

Debugging and Silent Invocation Options

This table describes the options which are common to all tools that the invoker can leverage.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbose</td>
<td>Verbose output</td>
</tr>
<tr>
<td></td>
<td>Log filled with Request/Response DataBinders</td>
</tr>
<tr>
<td>quiet</td>
<td>Minimal output</td>
</tr>
<tr>
<td>version</td>
<td>Print tool revision or version</td>
</tr>
<tr>
<td>log_file_name</td>
<td>Send program output to specified log file instead of the System.out log file</td>
</tr>
<tr>
<td>log_file_append</td>
<td>Append log to existing log file rather than overwrite it</td>
</tr>
<tr>
<td></td>
<td>Valid values: true, false</td>
</tr>
<tr>
<td></td>
<td>Default value: false</td>
</tr>
<tr>
<td>socketTimeout</td>
<td>Override time out of socket</td>
</tr>
<tr>
<td></td>
<td>Specify override time in seconds</td>
</tr>
</tbody>
</table>

You can use the tools to test the connection. Provide only the url, username, and password as you see in this sample test:

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic
password=we1com3i
```

Here you see the sample output:

Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing connection test (PING_SERVER) ...
Connection test successful.

Load Interface File for Import Process

Loads external setup or transaction data from the data file in the content repository to interface tables.

You run this process from the Scheduled Processes page. You can run it on a recurring basis.
Before running this process, you must:

1. Prepare your data file.
2. Transfer the data file to the content repository.

**Parameters**

**Import Process**
Select the target import process.

**Data file**
Enter the relative path and the file name of the ZIP data file in the content repository.

**Importing Data into Application Tables: Procedure**

The final destination for your external data is the application data tables of your Oracle Fusion Applications product.

Aspects of importing data into application tables involve the following:

- Loading data into interface tables
- Finding and submitting the import process

**Loading Data into Interface Tables**

Interface tables are intermediary tables that store your data temporarily while the system validates format and structure. Run the Load Interface File for Import scheduled process to load data from the data file into the interface table that corresponds to the template that you use to prepare the data.

To load your data into interface tables, submit the Load Interface File for Import scheduled process:

1. Sign in to Oracle Fusion Applications.
2. In the Navigator menu, select Tools, Scheduled Processes
3. Click the Schedule New Process button.
4. Search and select the Load Interface File for Import job.
5. When the Process Details page appears:
   a. Select the target import process.
   b. Enter the data file name.

**Note**

If the file exists in an account subordinate to the predefined account, you must enter the entire path relative to the predefined account in the content repository. Include all subaccounts and the file name.
6. Submit the process.

If no errors exist in the data file, then the process populates the interface tables.

---

**Note**
The data file remains in the content repository after the process ends.

---

**Finding and Submitting the Import Process**

Run the appropriate import process to import the data into the interface tables of your Oracle Fusion Applications product.

To import your data:

1. Sign in to Oracle Fusion Applications.
2. In the Navigator menu, select Tools, Scheduled Processes
3. Click the Schedule New Process button.
4. Find and select the import process that is specific to the target application tables.
5. When the Process Details page appears, select the process that corresponds to the data that you are importing.
   - If you prepared your data using the spreadsheet template, select the process named in the Overview section of the spreadsheet.
6. Submit the process.

---

**Note**
For more detailed information on the process used for data prepared using the spreadsheet template, see the Instructions and CSV Generation tab of the spreadsheet template.

---

**Correcting Import Load Process Errors: Explained**

The Load Interface File for Import process ends in error if the load of the data file fails on any row.

The following conditions apply when the process ends in error:

- The Load File to Interface child process ends in either warning or error.
- All rows that were loaded by the process are deleted, even those rows that loaded successfully.

To correct errors:

1. Review the error logs.
2. Change any formatting or structural anomalies that exist in the data.
3. Recreate the CSV and ZIP files.
4. Transfer the file to the content management server.
5. Submit the Load Interface File for Import job.
6. Repeat these steps until the process successfully loads the data.
7. Import the data using the appropriate product-specific process.
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 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.
- The Oracle Social Network business objects and the Navigator menu customizations are moved using the customization sets on the Customization Migration page instead of using the export and import function in the Setup and Maintenance work area.

<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 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>
</tbody>
</table>
| Application Attachment Category | Attachment categories and category-to-entity mappings | No parameters: all attachment categories and category-to-entity mappings are moved.  
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. |
|---------------------------------|-----------------------------------------------------|---------------------------------------------------------------------------------|
| Application Document Sequence Category | Document sequence categories | No parameters: all categories are moved.  
moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
code/applicationId: only the specified document sequence category code is moved. |
| 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.

**Note**
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.

| 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.

**Note**
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.
| Application Key Flexfield          | Key flexfield registration data and setup data | No parameters: all key flexfields are moved.  
|                                   |                                           | `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.  
| **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.  
| **Note**                          |                                           | Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.  
| Application Flexfield Value Set   | Value set setup data                      | No parameters: all value sets are moved.  
|                                   |                                           | `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.  
| **Note**                          |                                           | 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.  
| Application Reference Currency    | Currency data                             | No parameters: all currencies are moved.  
| Application Reference ISO Language| ISO language data                         | No parameters: all ISO languages are moved.  
| Application Reference Industry    | Industry data including industries in territories data | No parameters: all industries are moved.  

16-6 Oracle Fusion Marketing Implementing Marketing
<table>
<thead>
<tr>
<th>Application Reference Language</th>
<th>Language data</th>
<th>No parameters: all languages are moved.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Reference Natural Language</td>
<td>Natural language data</td>
<td>No parameters: all natural languages are moved.</td>
</tr>
<tr>
<td>Application Reference Territory</td>
<td>Territory data</td>
<td>No parameters: all territories are moved.</td>
</tr>
<tr>
<td>Application Reference Time zone</td>
<td>Time zone data</td>
<td>No parameters: all time zones are moved.</td>
</tr>
<tr>
<td>Application Standard Lookup</td>
<td>Standard lookup types and their lookup codes</td>
<td>No parameters: all standard lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Common Lookup</td>
<td>Common lookup types and their lookup codes</td>
<td>No parameters: all common lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Set-Enabled Lookup</td>
<td>Set-enabled lookup types and their lookup codes</td>
<td>No parameters: all set-enabled lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified set-enabled lookup is moved.</td>
</tr>
<tr>
<td>Application Profile Category</td>
<td>Profile categories</td>
<td>No parameters: all profile 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>name/applicationId: only the specified category is moved.</td>
</tr>
</tbody>
</table>
| 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      | Activity stream options | No parameters: all activity stream options are moved.  
| Configuration                    |                         |  
|                                  |                         |  

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.

address style format
Specifies the layout of an address, such as how many address lines it contains, and whether a city name is mandatory.

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 role
A role specific to applications and stored in the policy store.

Applications Core
Abbreviation for Oracle 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.
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.

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.

chart of accounts

The account structure your organization uses to record transactions and maintain account balances.

classification category

A grouping of classification codes, categorizing entities such as parties, tasks, and orders. Allows classification code assignment rules to be defined.

classification object

A special type of candidate object that 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.

condition

An XML filter or SQL predicate WHERE clause in a data security policy that specifies what portions of a database resource are secured.
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.

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 application’s 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.

**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.

**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 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 dimension 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.

**DMZ**

Acronym for demilitarized zone. An isolated internal network used for servers that are accessed by external clients on the Internet, such as web servers, to provide a measure of security for internal networks behind the firewall.

**do not contact**

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.

**document sequence**

A unique number that is automatically or manually assigned to a created and saved document.

**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.

**e-mail bounce**

An e-mail that is returned due to a temporary or permanent error condition.

**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**
Enterprise roles provide users with access both to the application functions they need to perform their jobs as well as the permissions to access the data where they need to perform those functions. There are two types of enterprise roles: job roles and abstract roles. Job roles permit users to perform activities specific to their job. Abstract roles permit users to perform functions that span the different jobs in the enterprise.

**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.

**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.

**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.

**hierarchy version**

Hierarchy version refers to a specific version of a hierarchy. You can create any number of versions, but only one hierarchy version can be active on a specific date. In tree terminology, hierarchy versions are also called tree versions.

**HTTP**

Acronym for Hypertext Transfer Protocol. A request and response standard typical of client-server computing. In HTTP, web browsers or spiders act as clients, while an application running on the computer hosting the web site acts as a server. The client, which submits HTTP requests, is also referred to as the user agent. The responding server, which stores or creates resources such as HTML files and images, may be called the origin server. In between the user agent and origin server may be several intermediaries, such as proxies, gateways, and tunnels.

**identifier type**

An identifier type is a specific category of additional identifiers, numeric or otherwise, which can identify trading community members. Examples of identifier types could be bank account numbers, passport numbers, and so on.

**interface table**

A database table used for transferring data between applications or from an external application or data file.

**Glossary-7**
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.

marketing activity
An instance of the execution of a campaign stage, such as delivering a specific treatment to a specific audience.

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.

MTA
Acronym for mail transfer agent. A software program that transfers electronic mail messages from one computer to another.

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.

name style format
Specifies the layout of a name, such as first name, last name, and phonetic last name.

name type
A name type is a specific category of additional names that can identify trading community members. Examples of name types could be maiden names, aliases, doing-business-as names, and so on.

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.

organization hierarchy
A tree structure that determines the relationships between organizations, such as which organizations are subordinate to other organizations.

overlay territory
A territory, usually owned by an internal employee, whose team supports the sales activities within the territory boundaries. Overlay territories often overlap with one or more prime or other overlay territories.

party
A physical entity, such as a person, organization or group, that the deploying company has an interest in tracking.
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.

person number
A person ID that is unique in the enterprise, allocated automatically or manually, and valid throughout the enterprise for all of a person's work and person-to-person relationships.

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.

prime territory
A territory that is usually owned by an internal employee who is directly responsible for sales within the territory boundaries. Prime territories aim to assign sales representatives to each region where potential customers are located.

privilege
A grant or entitlement of access to functions and data. A privilege is a single, real world action on a single business object.

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.

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.

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.

resource
People designated as able to be assigned to work objects, for example, service agents, sales managers, or partner contacts. A sales manager and partner contact can be assigned to work on a lead or opportunity. A service agent can be assigned to a service request.

resource organization
An organization whose members are resources. Resource organizations are used to implement sales organizations, partner organizations, and so on.

resource role
Resource roles indicate the role a resource plays as an individual, or within a resource team.
**resource team**

A resource team is a temporary group of resources formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization.

**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.

**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, payroll flows, and workforce business processes.

**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.

**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.

**SMTP**

Acronym for simple mail transfer protocol. A standard protocol used for sending e-mails across the Internet.

**SOA**

Abbreviation for service-oriented architecture.

**SOAP**

Acronym for simple object access protocol. A protocol specification, that relies on extensible markup language, for exchanging structured information in the implementation of web services within computer networks.

**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.
subscription confirmation 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.

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 into a hierarchy with one or more root nodes connected to branches of nodes. A tree must have a structure where each node corresponds to data from one or more data sources.

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 confirmation 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.

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 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.

**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.