Oracle Procurement Cloud
Implementing Procurement

This guide also applies to on-premise implementations

Release 8

April 2014
Contents

1 Getting Started with a Procurement Implementation

The Setup and Maintenance Work Area: Overview ...................................................... 1-1
Offerings: Explained ................................................................................................. 1-2
Options: Explained ................................................................................................. 1-2
Feature Choices: Explained ..................................................................................... 1-2
Implementation Projects: Explained ........................................................................ 1-3
Implementation Project Task Lists: Explained ....................................................... 1-4
Executing Setup Tasks: Explained ......................................................................... 1-4
Getting Started with an Implementation: Overview ............................................. 1-5
Generating a Procurement Setup Task List ............................................................... 1-6
Help Types: Explained ............................................................................................. 1-7

2 Common Applications Configuration: Define Synchronization of Users and Roles from LDAP

User and Role Synchronization: Explained ................................................................. 2-1

3 Common Applications Configuration: Define Implementation Users

Initial Security Administration: Critical Choices ...................................................... 3-1
Initial Security Administration: Worked Example .................................................. 3-3

4 Common Applications Configuration: Define Currencies and Currency Rates

Manage Currencies ..................................................................................................... 4-1
Manage Conversion Rate Types ............................................................................... 4-3
Manage Daily Rates .................................................................................................. 4-7

5 Common Applications Configuration: Define Enterprise Structures for Procurement

Enterprise Structures: Overview ............................................................................... 5-1
Enterprise Structures Business Process Model: Explained ...................................... 5-3
Global Enterprise Configuration: Points to Consider .............................................. 5-5
Modeling Your Enterprise Management Structure in Oracle Fusion: Example ........ 5-6
Essbase Character and Word Limitations ................................................................. 5-9
Define Initial Configuration with the Enterprise Structures Configurator ................ 5-11
Define Reference Data Sharing ............................................................................... 5-36
Define Enterprise: Manage Enterprise HCM Information ...................................... 5-43
Define Enterprise: Manage Locations ..................................................................... 5-44
Define Geographies ................................................................. 5-46
Define Legal Jurisdictions and Authorities ......................... 5-66
Define Legal Entities: Manage Legal Entity ......................... 5-72
Define Legal Entities: Manage Legal Entity HCM Information 5-77
Define Legal Entities: Manage Legal Entity Tax Profile .......... 5-82
Define Legal Entities: Define Legal Reporting Units .......... 5-86
Define Chart of Accounts for Enterprise Structures: Manage Chart of Accounts Structures and Structure Instances 5-87
Define Chart of Accounts for Enterprise Structures: Manage Chart of Accounts Value Sets and Value Set Values 5-108
Define Chart of Accounts for Enterprise Structures: Manage Accounting Calendars 5-111
Define Accounting Configurations of Enterprise Structures: Manage Primary or Secondary Ledgers 5-115
Define Accounting Configurations of Enterprise Structures: Specify Ledger Options 5-123
Define Accounting Configurations of Enterprise Structures: Manage Reporting Currencies 5-127
Define Business Units: Manage Business Units .................. 5-130
Define Business Units: Assign Business Unit Business Function 5-131
Define Business Units: Manage Service Provider Relationships 5-133
Define Business Units: Specify Supplier Contract Management Business Function Properties 5-139
Define Workforce Structures: Manage Locations .................. 5-141
Define Workforce Structures: Manage Divisions .................. 5-143
Define Workforce Structures: Manage Departments ............... 5-146
Define Workforce Structures: FAQs for Manage Job Families 5-149
Define Workforce Structures: Manage Job .......................... 5-150
Define Workforce Structures: Manage Person Search Relevance Profile Option Values 5-151
Define Facilities: Manage Facility Shifts, Workday Patterns, and Schedules 5-153
Define Facilities: Manage Inventory Organizations ............... 5-156
Define Facilities: Manage Item Organizations ..................... 5-161

6 Common Applications Configuration: Define Security for Procurement
Security Tasks: Highlights ....................................................... 6-1
Defining Security After Enterprise Setup: Points to Consider 6-4
Security Tasks and Oracle Fusion Applications: How They Fit Together 6-7
Security Tasks: Overview ....................................................... 6-10
Define Data Security .............................................................. 6-14
Define Users .................................................................. 6-32

7 Common Applications Configuration: Define Automated Governance, Risk, and Performance Controls
Segregation of Duties: Explained ............................................ 7-1
Segregation of Duties in the Security Reference Implementation: Explained 7-2
Defining Segregation of Duties Policies: Points To Consider 7-4
Managing Segregation of Duties Risks and Violations: Critical Choices .................................................. 7-6
Role Provisioning and Segregation of Duties: How They Work Together .............................................. 7-8

8 Common Applications Configuration: Define Approval Management for Procurement
Approval Management: Highlights ........................................................................................................... 8-1

9 Common Applications Configuration: Define Help Configuration
Define Help Configuration: Overview ...................................................................................................... 9-1
Set Help Options ..................................................................................................................................... 9-1
FAQs for Assign Help Text Administration Duty ..................................................................................... 9-4
Manage Help Security Groups ................................................................................................................ 9-4
Help File Customization ......................................................................................................................... 9-5
Embedded Help Customization .............................................................................................................. 9-16

10 Common Applications Configuration: Define Application Toolkit Configuration
Define Application Toolkit Configuration: Overview .................................................................................. 10-1
Map Reports to Work Areas .................................................................................................................... 10-1
Set Watchlist Options ............................................................................................................................. 10-3

11 Common Applications Configuration: Maintain Common Reference Objects
Maintain Common Reference Objects: Overview ...................................................................................... 11-1
Define Application Taxonomy .................................................................................................................. 11-1
Define Reference Data Sharing ............................................................................................................... 11-4
Define ISO Reference Data .................................................................................................................... 11-8
Manage Data Security Policies ................................................................................................................ 11-12
Set Activity Stream Options .................................................................................................................... 11-22
Manage Menu Customizations ................................................................................................................ 11-22
Manage Audit Policies ............................................................................................................................. 11-23
Manage Oracle Social Network Objects ................................................................................................. 11-25
Manage Applications Core Common Reference Objects ........................................................................ 11-29

12 Common Applications Configuration: Define WebLogic Communication Services
Configuration
Oracle Sales Cloud Computer Telephony Integration: Explained ......................................................... 12-1
Oracle Sales Cloud CTI: Top Tasks ........................................................................................................... 12-4
Configuring PSTN Gateway Address Using Topology Manager: Worked Example .................. 12-5

13 Define Applications Core Configuration
Define Lookups .......................................................................................................................................... 13-1
Manage Messages ..................................................................................................................................... 13-8
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define Document Sequences</td>
<td>13-10</td>
</tr>
<tr>
<td>Define Trees</td>
<td>13-13</td>
</tr>
<tr>
<td>Define Profile Options</td>
<td>13-29</td>
</tr>
<tr>
<td>Define Flexfields</td>
<td>13-38</td>
</tr>
<tr>
<td>Define Attachments</td>
<td>13-96</td>
</tr>
<tr>
<td><strong>14 Common Setup: Other Common Setup and Maintenance Tasks</strong></td>
<td></td>
</tr>
<tr>
<td>Define Transactional Business Intelligence Configuration</td>
<td>14-1</td>
</tr>
<tr>
<td>Define Extensions: Define Custom Enterprise Scheduler Jobs</td>
<td>14-4</td>
</tr>
<tr>
<td>Customization and Sandboxes</td>
<td>14-6</td>
</tr>
<tr>
<td><strong>15 Define Common Procurement Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Define Basic Catalogs</td>
<td>15-1</td>
</tr>
<tr>
<td>Define Supplier Configuration</td>
<td>15-19</td>
</tr>
<tr>
<td>Define Transaction Taxes</td>
<td>15-20</td>
</tr>
<tr>
<td>Manage Payment Terms</td>
<td>15-65</td>
</tr>
<tr>
<td>Manage Units of Measure</td>
<td>15-68</td>
</tr>
<tr>
<td>Define Corporate Procurement Cards</td>
<td>15-70</td>
</tr>
<tr>
<td>Define Common Payables and Procurement Options</td>
<td>15-71</td>
</tr>
<tr>
<td><strong>16 Define Purchasing Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Define Common Purchasing Configuration</td>
<td>16-1</td>
</tr>
<tr>
<td>Define Procurement Configuration Options</td>
<td>16-3</td>
</tr>
<tr>
<td>Configure Requisitioning Business Function</td>
<td>16-7</td>
</tr>
<tr>
<td>Define Procurement Agents</td>
<td>16-10</td>
</tr>
<tr>
<td><strong>17 Define Self Service Procurement Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Manage Information Template Descriptive Flexfields</td>
<td>17-1</td>
</tr>
<tr>
<td>Manage Catalog Category Hierarchy</td>
<td>17-3</td>
</tr>
<tr>
<td><strong>18 Define Supplier Portal Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Manage Supplier User Roles</td>
<td>18-1</td>
</tr>
<tr>
<td>Configure Supplier Registration</td>
<td>18-6</td>
</tr>
<tr>
<td>Specify Supplier News Content</td>
<td>18-8</td>
</tr>
<tr>
<td><strong>19 Define Sourcing Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Manage Negotiation Styles</td>
<td>19-1</td>
</tr>
<tr>
<td>Manage Attribute Lists</td>
<td>19-2</td>
</tr>
<tr>
<td>Manage Cost Factors</td>
<td>19-5</td>
</tr>
<tr>
<td>Manage Cost Factor Lists</td>
<td>19-6</td>
</tr>
</tbody>
</table>
20 Define Procurement Contracts Configuration

Define Contract Terms Library Configuration .......................................................... 20-1
Contract Terms Library Clauses: Explained ............................................................ 20-4
Contract Terms Templates: How They Work ....................................................... 20-6
Contract Expert Rules: How They Work ............................................................... 20-10

21 External Integration

Web Services: Overview .......................................................................................... 21-1
Files for Import and Export ................................................................................ 21-2
External Data Integration Services for Oracle Cloud .......................................... 21-6

22 Common Setup: Importing and Exporting Setup Data

Configuration Packages: Explained ....................................................................... 22-1
Exporting and Importing Setup Data: Explained ................................................. 22-1
Moving Common Reference Objects .................................................................... 22-2
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>
</tbody>
</table>

**Note**
Limited content applicable to Oracle Cloud implementations.

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.
Getting Started with a Procurement Implementation

The Setup and Maintenance Work Area: Overview

The Setup and Maintenance work area enables rapid and efficient planning, configuration, implementation, deployment, and ongoing maintenance of Oracle Fusion applications through self-service administration.

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

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

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

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

With Oracle Fusion Functional Setup Manager, the software behind the Setup and Maintenance work area, you can:

- Learn about and analyze implementation requirements.
- Configure Oracle Fusion applications to match your business needs.
- Get complete visibility to setup requirements through guided, sequential task lists downloadable into Excel for project planning.
- Enter setup data through easy-to-use user interfaces available directly from the task lists.
- Validate setup by reviewing setup data reports.
- Implement all Oracle Fusion applications through a standard and consistent process.
• Export and import data from one instance to another for rapid setup.

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.

### 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.

**Implementation Project Task Lists: Explained**

Once you make offering and functional area selections, Oracle Fusion Functional Setup Manager creates the implementation project and generates a complete list of setup tasks based upon your selections.

**Offering Top Task List**

A predefined hierarchical task list called the offering top task list is added. It includes a complete list of all tasks, including the prerequisites required to implement the offering. Typically, this task list will have the same name as the name of the offering it represents. If multiple offerings are included in a single implementation project, then each one of the offering top task lists is shown as a top node in the implementation task list hierarchy.

**Included Tasks**

Tasks needed to set up any of the dependent options and features, which are not selected for implementation are excluded from the task list. The implementation task list is generated according to the offering configurations and top task list definitions present at the time an implementation project is created. Once created, the task list in the implementation project becomes self-contained and will not change based on any changes made to the offering configurations or top task list definitions.

**Task Organization**

The offering top task list is shown as the top node in the implementation task list hierarchy. If multiple offerings are included in a single implementation project then top task list of each of the offerings becomes a top node of the implementation task list hierarchy. Within each top node, the tasks are organized with prerequisites and dependencies in mind.

- The most common requirements across all offerings are listed first.
- Next, the common tasks across an application area (such as Customer Relationship Management, or Financials), if applicable, are shown.
- Next, tasks that are common across multiple modules and options within an offering display.
- Finally, tasks for specific business areas of the offering, such as Opportunity Management, Lead Management, Territory Management, or Sales Forecasting are displayed.

**Executing Setup Tasks: Explained**

You enter setup data directly from the list of tasks in the Setup and Maintenance work area. You can find the task for which you want to enter data and then click
the corresponding button that allows you to go to the page where you perform the task. The page for managing setup data for the task will appear, where you enter data as appropriate. Once you finish entering data and close the page, you will return to the list of assigned implementation tasks. If an assigned setup task only uses a web service for managing its data, the web service will be executed when you perform the task.

**Note**

You cannot perform a task if you do not have the proper security entitlement.

You can add a file, URL, or text as notes to an assigned task. These notes will be accessible not only to you, but also to the implementation manager. All users assigned to the task will be able to see the notes.

**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.

Generating a Procurement Setup Task List

Create an implementation project to generate the setup tasks required for implementing the Oracle Fusion Procurement offering you have chosen for your implementation.
1. Use the Fusion Applications URL and provisioning super user ID and password to sign on to Fusion Applications. The Welcome page is displayed.

2. From the Navigator, select Setup and Maintenance under the Tools category. The Setup and Maintenance work area is displayed.

3. Configure Offerings:
   a. Select Configure Offering under Tasks
   b. Expand the Procurement offering to find its optional modules, called Options. Not all options are applicable to all businesses. A Fusion Applications implementation may include all, none, or any combination of the optional modules.
   c. Click the Enable for Implementation check box corresponding to the Procurement offering, then click the check box corresponding to the option.
   d. Click the Save and Close button.

4. Generate Setup Tasks:
   a. Select Manage Implementation Projects under Tasks.
   b. Click the Create button to create a new implementation project. This project will generate setup tasks for the offerings of your choice.
   c. Enter the following:
      • Name
      • Description
   d. Click the Next button

5. Review the auto generated Setup Task List
   a. Expand the list of tasks and check the Include checkbox for Procurement and the products you intend to configure.
   b. Click Save and Open Project. The generated task list is displayed
   c. Expand the task lists to see the relevant tasks

**Note**

The resulting Procurement Implementation Project will contain many tasks, but only a few tasks are required for setup. Each section in this guide refers to the task within the implementation project. If the task referenced in this guide is not present in the implementation project, access the task by using the All Tasks tab search on the Setup and Maintenance Overview page.

**Help Types: Explained**

Oracle Fusion Applications Help contains various types of help content, including demos, examples, FAQs, glossary terms, help topics, and PDF guides.
A business process or product can be supported by some or all of these help types.

**Demo**

Demos are Oracle User Productivity Kit (UPK) topics that visually demonstrate how to use the application to complete a short task or portion of a task. Demos can also provide an introduction to complex dashboards and work areas.

**Example**

Examples provide real use cases of features to illustrate how and when to use the feature, or scenarios to illustrate abstract concepts. Worked examples show exactly what you need to do to achieve a specific result, emphasizing decisions that you make and values that you enter.

**FAQ**

FAQs, or frequently asked questions, provide brief answers to questions that you might have regarding a task or page. For example, they can briefly explain what a term means, why something happened, how you can perform an action, or what happens if you perform the action.

**Glossary**

Glossary terms provide definitions for words or phrases used in help. You can search or browse glossary terms in the Glossary tab of Oracle Fusion Applications Help. Where the links are available, you can also see the definition when you hover over the term in help content for other help types.

**Help Topic**

Help topics explain key concepts, illustrate how application components work together, or assist in decision-making by explaining points to consider or the options you have. Help topics can also provide reference, overview, and other information.

**PDF Guide**

PDF guides in Oracle Fusion Applications Help provide, in a book format, information usually not found in other help types. There are other guides that present a collection of help content from the other help types, except demos, in an organized and logical format. These guides, for example, address specific business processes and setup offerings. You can find these guides by going to Navigator - Documentation Library in Oracle Fusion Applications Help.
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.
Common Applications Configuration: Define Implementation Users

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
- Defining 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>
<tr>
<td>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</td>
<td>Perform the Run User and Roles Synchronization Process task.</td>
</tr>
<tr>
<td>How to create a first implementation user</td>
<td>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.</td>
</tr>
<tr>
<td>How to establish security administration users</td>
<td>Define an IT security manager user provisioned with the IT Security Manager job role.</td>
</tr>
<tr>
<td>How to establish an implementation user with access to set up enterprise structures</td>
<td>Define an implementation user provisioned with the Application Implementation Consultant job role.</td>
</tr>
</tbody>
</table>

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 Inherit 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.

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.
Manage Currencies

Defining Currencies: Points to Consider

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

**Currency Codes**

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

**Date Ranges**

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

**Symbols**

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

**Euro Currency Derivation: Explained**

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

**Note**

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

**Derivation Type**

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

**Derivation Factor**

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

**Derivation Effective Date**

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

**FAQs for Manage Currencies**

**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.

---

**Manage Conversion Rate Types**

**Creating Conversion Rate Types: Critical Choices**

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

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

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

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

In creating new conversion rates, decide whether to do the following:
- Enforce inverse relationships
- Select pivot currencies
- Select contra currencies
- Enable cross rates and allow cross rate overrides
- Maintain cross rate rules

**Enforce Inverse Relationships**

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

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

**Select Pivot Currencies**

Select a pivot currency that is commonly used in your currency conversions. A pivot currency is the central currency that interacts with contra currencies. For example, you set up a daily rate between the US dollar (USD) and the Euro currency (EUR) and another between the USD and the Canadian dollar (CAD). USD is the pivot currency in creating a rate between EUR and CAD. EUR and CAD are the contra currencies. Select the pivot currency from the list of values which contains those currencies that are enabled, effective, 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
- Corporate
- User
Fixed

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

- Canadian dollar (CAD): A very stable currency
- Mexican Peso (MXP): A fluctuating currency
- Hong Kong dollar (HKD): An infrequently used currency

You enter two lines with accounts and amounts for each foreign currency transaction. Based on your company procedures, you select the appropriate rate type to populate the rate for Corporate and Spot rate types from your daily rates table. You manually enter the current rate for the User rate type.

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

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

FAQs for Manage Conversion Rate Types

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

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

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

## Manage Daily Rates

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

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

Every enterprise has three fundamental structures, legal, managerial, and functional, that are used to describe its operations and provide a basis for reporting. In Oracle Fusion, these structures are implemented using the chart of accounts and organizations. Although many alternative hierarchies can be implemented and used for reporting, you are likely to have one primary structure that organizes your business into divisions, business units, and departments aligned by your strategic objectives.
Legal Structure
The figure above shows a typical group of legal entities, operating various business and functional organizations. Your ability to buy and sell, own, and employ comes from your charter in the legal system. A corporation is a distinct legal entity from its owners and managers. The corporation is owned by its shareholders, who may be individuals or other corporations. There are many other kinds of legal entities, such as sole proprietorships, partnerships, and government agencies.

A legally recognized entity can own and trade assets and employ people in the jurisdiction in which it is registered. When granted these privileges, legal entities are also assigned responsibilities to:

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

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

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

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

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

**Enterprise Structures Business Process Model: Explained**

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

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

**Note**

There are product specific implementation activities that are not listed here and depend on the applications you are implementing. For example, you can

Global Enterprise Configuration: Points to Consider

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

- **Enterprise Configuration**
- **Business Unit Management**
- **Security Structure**
- **Compliance Requirements**

**Enterprise Configuration**

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

**Business Unit Management**

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

**Security Structure**

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

**Compliance Requirements**

How do you comply with your corporate external reporting requirements and local statutory reporting requirements? Do you tend to prefer a corporate first or
Modeling Your Enterprise Management Structure in Oracle Fusion: Example

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

Scenario

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

InFusion Corporation

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

Analysis

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

• Your company is required to report using US 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 Oracle Sales Cloud 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.
Essbase Character and Word Limitations

The following is a comprehensive list of character and word limitations that apply to Essbase. All of the limitations apply to all of the Oracle Fusion General Ledger configurations summarized in the table.

<table>
<thead>
<tr>
<th>Oracle Fusion General Ledger Configuration</th>
<th>Maps to Essbase As:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chart of Account Name</td>
<td>Cube Name</td>
</tr>
<tr>
<td>Chart of Account Segment Name</td>
<td>Dimension Name</td>
</tr>
<tr>
<td>Chart of Accounts Segment Value</td>
<td>Dimension Member Name</td>
</tr>
<tr>
<td>Chart of Accounts Segment Value Description</td>
<td>Alias for Member</td>
</tr>
<tr>
<td>Tree and Tree Version Name</td>
<td>Dimension Member Name</td>
</tr>
<tr>
<td>Primary Ledger Name</td>
<td>Dimension Member Name in Ledger Dimension</td>
</tr>
<tr>
<td>Secondary Ledger Name</td>
<td>Dimension Member Name in Ledger Dimension</td>
</tr>
<tr>
<td>Reporting Currency Name</td>
<td>Dimension Member Name in Ledger Dimension</td>
</tr>
<tr>
<td>Ledger Set Name</td>
<td>Dimension Member Name in Ledger Dimension</td>
</tr>
<tr>
<td>Accounting Calendar Period Names</td>
<td>Dimension Member Name in Accounting Period Name</td>
</tr>
<tr>
<td>Scenario Name Defined in Seeded Value Set Called Accounting Scenario</td>
<td>Dimension Member Name in Scenario Dimension</td>
</tr>
</tbody>
</table>

Even when case sensitivity is enabled in an aggregate storage outline for which duplicate member names is enabled, do not use matching names with only case differences for a dimension name. For example, do not:

- Name two dimensions Product and product.
- Use quotation marks or brackets.
- Use tabs in dimension, member, or alias names.
- Use accent characters.
- Use the characters for dimension or member names.

**Restricted Characters**

The following is a list of characters that are restricted and can not be used in dimension, member, or alias names.

<table>
<thead>
<tr>
<th>Character</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>@</td>
<td>at sign</td>
</tr>
<tr>
<td>\</td>
<td>backslash</td>
</tr>
<tr>
<td>,</td>
<td>comma</td>
</tr>
<tr>
<td>-</td>
<td>dash, hyphen, or minus sign</td>
</tr>
<tr>
<td>=</td>
<td>equal sign</td>
</tr>
<tr>
<td>&lt;</td>
<td>less than sign</td>
</tr>
<tr>
<td>(</td>
<td>parentheses</td>
</tr>
</tbody>
</table>
Other Restrictions

- Do not place spaces at the beginning or end of names. Essbase ignores such spaces.

- Do not use these types of words as dimension or member names:
  - Calculation script commands, operators, and keywords.
  - Report writer commands.
  - Function names and function arguments.
  - Names of other dimensions and members (unless the member is shared).
  - Generation names, level names, and aliases in the database.

- Any of these words in the table below:

<table>
<thead>
<tr>
<th>List 1</th>
<th>List 2</th>
<th>List 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>AND</td>
<td>ASSIGN</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>CALC</td>
<td>CALCMBR</td>
</tr>
<tr>
<td>COPYFORWARD</td>
<td>CROSSDIM</td>
<td>CURMBRNAME</td>
</tr>
<tr>
<td>DIM</td>
<td>DIMNAME</td>
<td>DIV</td>
</tr>
<tr>
<td>DYNAMIC</td>
<td>EMPTYPARM</td>
<td>EQ</td>
</tr>
<tr>
<td>EQOP</td>
<td>EXCEPT</td>
<td>EXP</td>
</tr>
<tr>
<td>EXPERROR</td>
<td>FLOAT</td>
<td>FUNCTION</td>
</tr>
<tr>
<td>GE</td>
<td>GEN</td>
<td>GENRANGE</td>
</tr>
<tr>
<td>GROUP</td>
<td>GT</td>
<td>ID</td>
</tr>
<tr>
<td>IDERROR</td>
<td>INTEGER</td>
<td>LE</td>
</tr>
<tr>
<td>LEVELRANGE</td>
<td>LOOPBLOCK</td>
<td>LOOPPARMS</td>
</tr>
<tr>
<td>LT</td>
<td>MBR</td>
<td>MBRNAME</td>
</tr>
<tr>
<td>MBRONLY</td>
<td>MINUS</td>
<td>MISSING, #MISSING</td>
</tr>
<tr>
<td>MUL</td>
<td>MULOP</td>
<td>NE</td>
</tr>
<tr>
<td>NON</td>
<td>NONINPUT</td>
<td>NOT</td>
</tr>
<tr>
<td>OR</td>
<td>PAREN</td>
<td>PARENPARM</td>
</tr>
<tr>
<td>PERCENT</td>
<td>PLUS</td>
<td>RELOP</td>
</tr>
<tr>
<td>SET</td>
<td>SKIPBOTH</td>
<td>SKIPMISSING</td>
</tr>
<tr>
<td>SKIPNONE</td>
<td>SKIPZERO</td>
<td>TO</td>
</tr>
<tr>
<td>TOLOCALRATE</td>
<td>TRAILMISSING</td>
<td>TRAILSUM</td>
</tr>
</tbody>
</table>
Define Initial Configuration with the Enterprise Structures Configurator

Establishing Enterprise Structures Using the Enterprise Structures Configurator: Explained

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

This figure illustrates the process to configure your enterprise using the Enterprise Structures Configurator.

To be able to use the Enterprise Structures Configurator, you must select the Enterprise Structures Guided Flow feature 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>
<tbody>
<tr>
<td>Country</td>
<td>• US</td>
</tr>
<tr>
<td></td>
<td>• UK</td>
</tr>
<tr>
<td></td>
<td>• Japan</td>
</tr>
<tr>
<td></td>
<td>• India</td>
</tr>
<tr>
<td>Country and Division</td>
<td>• InFusion Lighting: Japan</td>
</tr>
<tr>
<td></td>
<td>• InFusion Lighting: US</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: UK</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: India</td>
</tr>
<tr>
<td>Country and business function</td>
<td>• Sales: Japan</td>
</tr>
<tr>
<td></td>
<td>• Marketing: Japan</td>
</tr>
<tr>
<td></td>
<td>• Sales: US</td>
</tr>
<tr>
<td></td>
<td>• Sales: UK</td>
</tr>
<tr>
<td></td>
<td>• Marketing: India</td>
</tr>
<tr>
<td></td>
<td>• Sales: India</td>
</tr>
<tr>
<td>Division</td>
<td>• InFusion Lighting</td>
</tr>
<tr>
<td></td>
<td>• InFusion Security</td>
</tr>
<tr>
<td>Division and Legal Entity</td>
<td>• InFusion Lighting: Japan</td>
</tr>
<tr>
<td></td>
<td>• InFusion Lighting: US</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: UK</td>
</tr>
<tr>
<td></td>
<td>• Infusion Security: India</td>
</tr>
</tbody>
</table>
Manually Creating Business Units

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

Reference Data Sets and Sharing Methods: Explained

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

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

Reference Data Sets

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

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

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

Reference Data Sharing Methods

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

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

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

- Assignment to multiple sets, no common values allowed. The method of sharing reference data that allows a reference data object instance to be assigned to multiple sets. For example, 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.
Reference data sharing is a feature within Oracle Fusion that enables you to group set-enabled reference data such as jobs or grades so that the data can be shared across different parts of the organization. Sets also enable you to filter reference data at the transaction level so that only data that has been assigned to certain sets is available to select. To filter reference data, Oracle Fusion Human Capital Management (HCM), applications use the business unit on the transaction. To set up reference data sharing in Oracle Fusion HCM, you create business units and sets, and then assign the sets to the business units.

**Common Set Versus Specific Sets**

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

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

**Business Unit Set Assignment**

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

**Example: Assigning Sets to Business Units**

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

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

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

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

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

InFusion assigns business units to sets as follows:

<table>
<thead>
<tr>
<th>Business Unit</th>
<th>Default Set Assignment</th>
<th>Set Assignment Overrides</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales_Japan</td>
<td>Sales_Japan_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
<tr>
<td>Marketing_Japan</td>
<td>Mktg_Japan_Set for grades, departments, and locations</td>
<td>None</td>
</tr>
<tr>
<td>Sales_US</td>
<td>Sales_US_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
<tr>
<td>Sales_UK</td>
<td>Sales_UK_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
<tr>
<td>Marketing_India</td>
<td>Mktg_India_Set for grades, departments, and locations</td>
<td>None</td>
</tr>
<tr>
<td>Sales_India</td>
<td>Sales_India_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
</tbody>
</table>

When setting up grades, departments, and locations for the business units, InFusion will assign the data to the default set for each business unit. When setting up jobs, they will assign the Jobs set and will assign the Common Set to any jobs that may be used throughout the entire organization.

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

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

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

Creating Reference Data Sets in the Enterprise Structures Configurator: Explained

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

A standard set called the Enterprise set is predefined.

Common Set

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

Jobs and Positions: Critical Choices

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

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

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

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

**Primary Industry of Your Enterprise**

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

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

**Management of People**

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

**Positions: Examples**

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

**Retail Industry**

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

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

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

This figure illustrates the retail position setup.
Health Care Industry

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

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

This figure illustrates the hospital position setup.

Jobs: Example

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

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

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

This figure illustrates the software industry job setup.

Job and Position Structures: Explained

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

Enterprise-Level Job Attributes

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

Enterprise-Level Position Attributes

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

**What's the default reference data set?**

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

**What happens if I override the set assignment?**

For the selected business unit, you can override the default reference data set for one or more reference data groups. For example, assume you have three reference data groups: Vision 1 SET, Vision 2 SET, and Vision 3 SET, where Vision
SET 1 is the default set for business unit United Kingdom Vision 1 BU. You can override the default so that grades are assigned to Vision 2 SET, departments are assigned to Vision 3 SET, and jobs are assigned to the default set, Vision 3 SET.

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.

![Diagram of reference data sharing methods](image)

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

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

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

**Reference Data Sets**

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

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

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

**Reference Data Sharing Methods**

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

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

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

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

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

Assigning Reference Data Sets to Reference Objects: Points to Consider

You can assign the reference data sets to reference objects on the Manage Reference Data Set Assignments page. For multiple assignments, you can classify different types of reference data sets into groups and assign them to reference entity objects. The assignment takes into consideration the determinant type, determinant, and reference group, if any.

Determinant Types

The partitioned reference data is shared based on a business context setting called the determinant type. It is the point of reference used in the data assignment process. The following table lists the determinant types used in the reference data assignment.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset Book</td>
<td>Information about the acquisition, depreciation, and retirement of an asset that belongs to a ledger or a business unit.</td>
</tr>
<tr>
<td>Business Unit</td>
<td>The departments or organizations within an enterprise.</td>
</tr>
<tr>
<td>Cost Organization</td>
<td>The organization used for cost accounting and reporting on various inventory and cost centers within an enterprise.</td>
</tr>
<tr>
<td>Project Unit</td>
<td>A logical organization within an enterprise that is responsible for enforcing consistent project management practices.</td>
</tr>
<tr>
<td>Reference Data Set</td>
<td>References to other shared reference data sets.</td>
</tr>
</tbody>
</table>

Determinant

The determinant or determinant value is the value that corresponds to the selected determinant type. The determinant is one of the criteria for selecting the appropriate reference data set. For example, when managing set assignments for the set determinant type, Reference Data Set is the determinant type, and you would enter the corresponding set code value as the corresponding determinant value.

Reference Groups

A transactional entity may have multiple reference entities (generally considered to be setup data) that are treated in the same manner because of commonness in
implementing business policies and legal rules. Such reference entities in your application are grouped into logical units called reference groups, based on the functional area and the partitioning requirements that they have in common. For example, all tables and views that define Sales Order Type details might be part of the same reference group.

**Note**

The reference groups are predefined in the reference groups table and are available for selection and assignment.

---

### Items and Supplier Site Reference Data Sharing: Explained

Some products required special logic for reference data sharing and have implemented their own domain specific ways for sharing data.

Items and supplier sites are two such product specific reference data objects that use product specific mechanisms to share data.

**Items**

If you share your items across warehouses or manufacturing facilities, you can access them through a common item master. Configure one or multiple item masters for your enterprise, based your enterprise structure. A single item master is recommended because it provides simpler and more efficient maintenance. However, in rare cases, it may be beneficial to keep multiple item masters. For example, if you acquire another enterprise and need to continue to operate your lines of business separately, maintaining a second item master might be the best decision.

**Suppliers Sites**

You can approve particular suppliers to supply specified commodities and authorize your business units to buy from those suppliers when the need arises. For example, you might be a household cleaning products manufacturer and need dyes, plastics, and perfumes to make your products. You purchase from a central supplier 70% of your perfume supplies with an additional supplier, in reserve, from whom you purchase the remaining 30%. At the same time, each of your business units purchases plastics and dyes from the same supplier, but from different local supplier sites to save transportation costs.

To implement business unit specific supplier sites, Oracle Fusion Procurement supports a method for defining suppliers sites as owned and managed by the business unit responsible for negotiating the supplier terms. Your other business units that have a service provider relationship defined with your procurement business unit, subscribe to the supplier sites using the supplier site assignments feature. In addition, Procurement allows sharing of the following procurement data objects across business units:

- Supplier qualification data, such as approved supplier lists
- Catalog content, such as agreements, smart forms, public shopping lists, and content zones
- Procurement configuration data
FAQs for Define Reference Data Sharing

What reference data objects can be shared across business units?

The following list contains the reference data objects for the Oracle Fusion Applications that can be shared across business units and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trading Community Model</td>
<td>Customer Account Relationship</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Trading Community Model</td>
<td>Customer Account Site</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Trading Community Model</td>
<td>Sales Person</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Opportunity Management</td>
<td>Sales Method Group</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Work Management</td>
<td>Assessment Templates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Enterprise Contracts</td>
<td>Contract Types</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Sales</td>
<td>Sales Method</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Common Components</td>
<td>Activity Templates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Payables</td>
<td>Payment Terms</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
<tr>
<td>Receivables</td>
<td>Accounting Rules</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Aging Buckets</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Auto Cash Rules</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Collectors</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Lockbox</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Memo Lines</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Payment Terms</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Remit To Address</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Revenue Contingencies</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Receivables</td>
<td>Transaction Source</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Application Name</td>
<td>Reference Data Object</td>
<td>Method of Sharing</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>Receivables</td>
<td>Transaction Type</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Advanced Collections</td>
<td>Collections Setups</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Advanced Collections</td>
<td>Dunning Plans</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Tax</td>
<td>Tax Classification Codes</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Departments</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Jobs</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Locations</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Human Resources</td>
<td>Grades</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Project Billing</td>
<td>Project and Contract Billing</td>
<td>Assignment to multiple sets, common values not allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Accounting Definition</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Rates</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Distributed Order Orchestration</td>
<td>Hold Codes</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Distributed Order Orchestration</td>
<td>Orchestration Process</td>
<td>Assignment to one set only, with common values</td>
</tr>
</tbody>
</table>

**What reference data objects can be shared across asset books?**

The following list contains the reference data objects for Oracle Fusion Assets that can be shared across asset books and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets</td>
<td>Bonus Rules</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Depreciation Ceilings</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Depreciation Methods</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Assets</td>
<td>Asset Descriptions</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Property Types</td>
<td>Assignment to one set only, with common values</td>
</tr>
<tr>
<td>Assets</td>
<td>Prorate Conventions</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
<tr>
<td>Assets</td>
<td>Asset Queue Names</td>
<td>Assignment to one set only, with common values</td>
</tr>
</tbody>
</table>
### What reference data objects can be shared across cost organizations?

The following list contains the reference data objects for Oracle Fusion Cost Management that can be shared across cost organizations and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost Management</td>
<td>Cost Structure</td>
<td>Assignment to one set only, no common values allowed</td>
</tr>
</tbody>
</table>

### What reference data objects can be shared across project units?

The following list contains the reference data objects for Oracle Fusion Project Foundation that can be shared across project units and the method in which the reference data for each is shared.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Reference Data Object</th>
<th>Method of Sharing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Foundation</td>
<td>Project Definition</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
<tr>
<td>Project Foundation</td>
<td>Project Transaction Types</td>
<td>Assignment to multiple sets, no common values allowed</td>
</tr>
</tbody>
</table>

### Define Enterprise: Manage Enterprise HCM Information

**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.

**Define Enterprise: Manage Locations**

**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.

FAQs for Manage Locations

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

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

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

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

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

The location is available for selection in purchase documents of that inventory organization in Oracle Fusion Inventory Management. If you don't select an inventory organization, then the location is available in purchase documents across all inventory organizations.
**What happens if I inactivate a location?**

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

**How can I associate a location with an inventory organization?**

From the Manage Locations page in Oracle Fusion Global Human Resources. To appear on the Create or Edit Location pages, your inventory organization must be effective on today’s date and must exist in the location set that you selected.

**Define Geographies**

**Defining Address Cleansing: Explained**

Address cleansing provides a way to validate, correct, and standardize addresses that are entered in a user interface. Geography validation only validates the geography attributes of an address, for example, State, City, and Postal codes; address cleansing validates both the geography attributes and the address line attributes.

To be able to use the address cleansing functionality, you need to have license for the customer data quality application, because the feature is delivered using data quality integration.

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 the application does not find a matching address it will alert you.

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

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

**Geography Structure**

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

**Geography Hierarchy**

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

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

**Geography Validation**

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

**Geography Structures: Explained**

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

You can use the geography structure to establish:

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

**How Geographies Can Be Related**

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

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

**Note**

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

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

**The Types of Geographies You Can Define for the Country**

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

**Note**

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

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

**Geography Hierarchy: Explained**

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

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

The application leverages geography hierarchy information to facilitate business processes that rely on geography information, for example, tax calculation, order sourcing rules, sales territory definition. The geography hierarchy information is centrally located 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 comprises all geography related data, including geography types and geographies.

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.

Importing Geographies: Explained

A geography, such as Tokyo or Peru, describes a boundary on the surface of the earth. You can create new geographies by importing data through interface tables. There are two options for populating the interface tables: using the tool of your preference to load the data or using file-based data import. If you plan to provide the data details in a source file, use the file-based import feature. If you will populate the interface table directly, run the geography loader process to import the data. Having a good understanding of the import entity, interface table, and destination table will help you prepare your import data.

Consider the following when importing geographies:

- File-based import option
- Geography loader process option
- Import object entity, interface table, and destination tables

**File-Based Import Option**

The file-based import process reads the data included in your XML or text file, populates the interface tables, and imports the data into the application destination tables. The **File-Based Data Import Setup and Maintenance** task list includes the tasks needed to configure the geography import object, create source file mappings, and schedule the import activities.

**Geography Loader Process Option**

Populate the interface table with your import data, then navigate to the **Run Geography Loader Setup and Maintenance** task to schedule the import of data from the interface table to the destination table.

**Import Object Entity, Interface Table, and Destination Tables**

The geography import object consists of one entity and interface table that forms the geography. If you are using file-based import, you can map your source file data to import entity attributes that correspond to the interface table columns. The import activity process populates the interface table based on the mapping and your source file. If using the geography loader scheduled process, populate the interface table directly using your preferred tool. If you need the unique IDs of existing application data for your import data, use the **Define Data Export Setup and Maintenance** task list to export the information.

**Note**

Spreadsheets containing detailed information about each interface table, including the import attributes, corresponding interface table columns, defaults, and validations, are available from the Oracle Enterprise Repository by searching
on a specific interface table name or initiating a search using the FusionApps: Interface Table asset type.

The following lists the object entity, tables, and resulting application object:

<table>
<thead>
<tr>
<th>File-Based Import Entities</th>
<th>Interface Tables</th>
<th>Destination Tables</th>
<th>Application Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>HZ_IMP_GEOGRAPHIES</td>
<td>HZ_GEOGRAPHIES</td>
<td>Geography</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_GEOGRAPHY_IDEN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_GEOGRAPHY_TYPE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_HIERARCHY_NODE</td>
<td></td>
</tr>
</tbody>
</table>

**Importing Country Structures Using File-Based Import: Explained**

This topic explains how to prepare and import country structure data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. A country structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States has the geography type of State at the top, followed by the County, then the City, and finally the Postal Code.

You can use the country structure to set up the following:

- The relationships between geographies within a country
- The types of geographies that you can define for a country

Consider the following questions when importing your data:

- How does your legacy system or source system represent the country structure compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map to your data values?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

**Comparing Business Object Structures**

You must understand how your country structure data corresponds with the data in Oracle Sales Cloud in order to be able to map your legacy data to the data needed by Oracle Sales Cloud. First, you must understand how Oracle Sales Cloud represents the structure of the data for a country structure.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country’s structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.
Import Objects for the Country Structure

To facilitate the import of country structures, Oracle Sales Cloud incorporates the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the country structure.

A good understanding of the attribute details of the import objects is critical to preparing your import data. For information about the Oracle Sales Cloud attributes, see the Oracle Enterprise Repository. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Sales Cloud attributes. The validation information includes the navigation to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>Country Structure Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Extensible Attributes

If you need to extend the Oracle Sales Cloud object to import your legacy or source data, you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Country Structures Using File-Based Data Import

For the country structure business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new country structure, you import the Country Structure object.

You must be assigned the Master Data Management Administrator job role to access and submit the import activities for country structures.
Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity.

Country Structure Import Objects: How They Work Together

This topic describes the Country Structure import object. You use the Country Structure import object when you submit a file-based import activity to import your country structure information. This topic introduces the following:

- Target import object concepts
- Target objects for the Country Structure import object
- Target import object attributes
- Target object attribute reference guide files

Target Import Object Concepts

The Country Structure import object is used to import a country structure hierarchy, including details, such as geography type, geography type name, parent geography type, geography level numbers, and so on. To map the source data in your import file to the target attributes in Oracle Sales Cloud, you must understand how the target objects are related and what attributes are included in each target object.

Country Structure Target Import Objects

The Country Structure import object contains one target import object that organizes the individual attributes of the different aspects of the geography structure. When updating an existing country structure, you must provide the parent reference information of the existing country structure. This reference information connects the imported geography structure to the existing one. Use the ImpGeoStructureLevel target import object to create and update country structure information.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.
Note

If any of the attributes you want to import does not have an equivalent target object attribute, then review the Application Composer extensibility features for country structures.

Target Import Objects Attributes Resources

To access the reference guide files for the country code’s target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository (http://fusionappsor.oracle.com).

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference guide files that are available from the Documentation tab for the Country Code File-Based Data Import asset.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeoStructureLevel</td>
<td>Contains information that specifies a country’s geography structure.</td>
<td>HZ_IMP_GEO_STRUCTURE_LEVELS_Reference</td>
</tr>
<tr>
<td></td>
<td>Sample attributes: GeographyType, GeographyTypeName, LevelNumber, and ParentGeographyType.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Reference attribute: CountryCode</td>
<td></td>
</tr>
</tbody>
</table>

Importing Geographies Using File-Based Import: Explained

This topic describes the tasks you must perform to import geography information. A geography is any region with a boundary around it, regardless of its size. It might be a state, a country, a city, a county, or a ward. You must create or import geographies before you can associate them with custom zones and addresses.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map to your data values?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your geography data corresponds with the data in Oracle Sales Cloud in order to be able to map your legacy data to the data needed by Oracle Sales Cloud. First, you must understand how Oracle Sales Cloud represents the structure of the data for a geography.
You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country’s structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

**Import Objects for the Geography**

To facilitate the import of geographies, Oracle Sales Cloud incorporates the structure of the geography into import objects. The import object for the geography is ImpGeography.

**Comparing Business Object Data**

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the geography.

A good understanding of the attribute details of the import objects is critical to preparing your import data. For information about the Oracle Sales Cloud attributes, see the Oracle Enterprise Repository. The reference guide files contain descriptions, logic used to choose default values, and validation information for each import object attribute. The validation information includes the navigation to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>Geography Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Hint: You can use the keyword importing geographies to search for related topics in Help.

**Extensible Attributes**

Oracle Sales Cloud does not support extensible attributes for geographies. You can only import data for attributes provided by Oracle Sales Cloud.

**Importing Geographies Using File-Based Data Import**

For the geography business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new geography, you import the Geography
You must be assigned the Master Data Management Administrator job role to access and submit the import activities for geographies. When importing geography information, you must provide the parent reference information for all parent levels for the entity.

**Verifying Your Imported Data**

Oracle Sales Cloud provides File-Based Import activity reports, which can be used to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Geographies work area to view the imported geographies.

**Geography Import Objects: How They Work Together**

This topic describes the Geography import object. You use the Geography import object to import geography information.

This topic introduces the following:

- Target import object concepts
- Target objects for the Geography import object
- Target import object attributes
- Target import object attribute reference guide files

**Target Import Object Concepts**

The Geography import object is used to import geography hierarchy information to create or update the geography data of a country. To map the source data in your import file to the target attributes in Oracle Sales Cloud, you must understand how the target objects are related and what attributes are included in each target object.

**Geography Target Import Objects**

The target import objects in the Geography import object contain information about the geography hierarchy. When updating an existing geography, you must provide the parent reference information of the existing geography, which connects the geography to the country of which it is a part.

Use the ImpGeography target import object to create and update geography information.

**Note**

Before you import geography data for a country, you must define the country’s geography structure.

**Target Import Objects Attributes**

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each
attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

**Target Import Objects Attributes Resources**

To access the reference guide files for the geography’s target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository (http://fusionappsoer.oracle.com).

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference guide files that are available from the Documentation tab for the Geography File-Based Data Import asset.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Attribute Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>Contains information that captures a country’s geography hierarchy details. Sample attributes: CountryCode, GeoDataProvider, GeographyType, PrimaryGeographyCode, PrimaryGeographyCodeType, and PrimaryGeographyName. Reference attribute: CountryCode</td>
<td>HZ_IMP_GEOGRAPHIES_T_Reference</td>
</tr>
</tbody>
</table>

**Importing Geographies Using File-based Data Import: Worked Example**

This example demonstrates how to import data using the File-Based Data Import tool. In this particular example, you have a source file containing geography data that you want to import into the application, so that the geography data can be used for real time address validation and tax purposes.

The following table summarizes the key decisions that you need to make in this scenario:

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What type of object are you importing?</td>
<td>Geography</td>
</tr>
<tr>
<td>What file type are you using for your source data?</td>
<td>Text file</td>
</tr>
<tr>
<td>Where are you uploading your source data file from?</td>
<td>Your desktop</td>
</tr>
<tr>
<td>What data type is your source data file?</td>
<td>Comma separated</td>
</tr>
</tbody>
</table>
Which fields are you importing into Oracle Sales Cloud? | All, except for the RecordTypeCode field
---|---
When do you want to process the import? | Immediately

**Summary of the Tasks**

These are the steps that are required to create an import activity and activate the import:

1. Determine what information is in the source file.
2. Create and schedule the import activity.
3. Monitor the import results.

**Prerequisites When Importing Additional Geography Data After Your Initial Import**

1. You need to ensure that the combination of Source ID and Parent Source ID values are unique for each row of data within a single import. However, your source data files do not need to have the same Source ID and Parent Source ID values as your previously imported geography data. If the geography structure levels and the parents for each geography value are the same, the changed IDs will not affect the import.

2. Ensure that all of the parents of a child geography are included in your data file so that the child geography can be added. For example, if you originally imported US, CA, and San Francisco, and now you want to import the city of San Jose in CA, then your data file needs to include US, CA, and San Jose.

3. Check that your source data file has the correct values for the geography data that you have already loaded. For example, if your initial import included the value US for country and CA as state, and in a subsequent import you have California as a state, your geography import will result in two state records (CA and California) in the application data, with the US as the country parent.

**Determine What Information Is in the Source File**

1. Your source geography data files should include a unique Source ID value for each row of data, and a Parent Source ID value which identifies the parent of that row of geography data. Source IDs, or Parent Source IDs, should not be longer than 18 characters. You could structure your geography source data as follows:

<table>
<thead>
<tr>
<th>Geography Level</th>
<th>Name</th>
<th>Source ID</th>
<th>Parent Source ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (Country)</td>
<td>US</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2 (State)</td>
<td>CA</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>3 (County)</td>
<td>Alameda</td>
<td>111</td>
<td>11</td>
</tr>
<tr>
<td>4 (City)</td>
<td>Pleasanton</td>
<td>1111</td>
<td>111</td>
</tr>
<tr>
<td>4 (City)</td>
<td>Dublin</td>
<td>1112</td>
<td>111</td>
</tr>
</tbody>
</table>
Create and Schedule the Import Activity

You create an import activity, enter the import details, and schedule the import. An import activity definition provides instructions for the import processing, including details related to selecting the source file, or file location; mapping fields from the source file to the Oracle Sales Cloud database object and attribute; and scheduling the import.

1. Navigate to Setup and Maintenance and search for the Manage File Import Activities task. Click Go to Task.

2. In the Manage Import Activities page, click the Create icon.

3. In the Create Import Activity: Set Up page, create an import activity for the Geography object type by completing the fields, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Master Reference Geographies</td>
</tr>
<tr>
<td>Object</td>
<td>Geography</td>
</tr>
<tr>
<td>File Type</td>
<td>Text File</td>
</tr>
<tr>
<td>File Selection</td>
<td>Specific file</td>
</tr>
<tr>
<td>Upload From</td>
<td>Desktop</td>
</tr>
<tr>
<td>File Name</td>
<td>Choose relevant file from desktop</td>
</tr>
<tr>
<td>Data Type</td>
<td>Comma separated</td>
</tr>
</tbody>
</table>

Note

Ensure that the file type that you select in the Create Import Activity: Set Up page matches the file type of the source data file.

4. Click Next.

5. On the Create Import Activity: Map Fields page, map each field from your source file to the Oracle Sales Cloud database object and attribute, as shown in this example:

<table>
<thead>
<tr>
<th>Column Header</th>
<th>Example Value</th>
<th>Ignore</th>
<th>Object</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Geography Name</td>
<td>Primary Geography Name</td>
<td>United States</td>
<td>Imp Geography</td>
<td>Primary Geography Name</td>
</tr>
<tr>
<td>Country Code</td>
<td>US</td>
<td>No</td>
<td>Imp Geography</td>
<td>Country Code</td>
</tr>
<tr>
<td>Record Type Code</td>
<td>0</td>
<td>Yes</td>
<td>Imp Geography</td>
<td>Record Type Code</td>
</tr>
<tr>
<td>Source ID</td>
<td>10265</td>
<td>No</td>
<td>Imp Geography</td>
<td>Source ID</td>
</tr>
<tr>
<td>Parent Source ID</td>
<td>1053</td>
<td>No</td>
<td>Imp Geography</td>
<td>Parent Source ID</td>
</tr>
</tbody>
</table>
If you do not want to import a column in the text file you can select **Ignore**.

**Note**

If you have any difficulties mapping the fields from your source file to the relevant Oracle Sales Cloud database object, you can use the import object spreadsheets for reference.

6. Click **Next**.

7. On the Create Import Activity: Create Schedule page, select **Immediate** in the Schedule field so that the import will start as soon as you activate it.

Instead of immediately importing the data, you can choose a date and time to start the import. You can also specify if the import will be repeated, and the frequency of the repeated import.

8. Click **Next**.

**Monitor the Import Results**

You monitor the progress of the import activity processing, and view completion reports for both successful records and errors.

1. On the Create Import Activity: Review and Activate page, verify your import details in the Import Details, File Details, Import Options, and Schedule sections. Update the import details if required by navigating to the previous screens using the **Back** link.

2. Once you are sure your import details are correct, click **Activate** to submit the import.

Once the import activity has completed, the Status field value will change to **Completed**.

**Importing and Exporting Territory Geography Zones: Explained**

Territory geography zones are geographical boundaries that you can set up to replicate your organization's regions, such as a Pacific Northwest sales region. You can set up territory geography zones in one Oracle Sales Cloud applications instance, and then after the territory geography zones are defined you can export the territory zones and import them into another Oracle Sales Cloud instance.

To define your territory geography zones and then import your territory zones into another Oracle Sales Cloud instance, you need to complete the following steps:

1. Import the master reference geography data into the Oracle Sales Cloud.
2. Define your territory geography zones using the Manage Territory Geographies task.
3. Export the territory geography zones.
4. Import the territory geography zones into another Oracle Sales Cloud instance.

**Import the master reference geography data**

Firstly, you need to import the master reference geography data. Master reference geography data consists of geography elements such as country, state, and city, and is required for any geographical information you store in the application, such as address information used in customer and sales records. For more information, refer to the Geography Hierarchy: Explained topic listed in the related topics section. Master reference geography data can be imported into the application using the Manage File Import Activities task in Setup and Maintenance - refer to the Importing Master Reference Geography Data: Worked Example topic listed in the related topics section for more information.

**Define your territory geography zones**

Once the master reference geography data has been imported, you can then create your territory geography zones in the application using the Manage Territory Geographies task in Setup and Maintenance. For more information, refer to the Managing Territory Geographies: Worked Example topic listed in the related topics section.

**Export the territory geography zones**

Once you have completed importing the master reference geography data and defining your territory geography zone tasks, you can create a configuration package to export the territory zone data. For more information, refer to the Exporting Setup Data demo listed in the related topics section.

**Import the territory geography zones**

Once you have downloaded your configuration package for your territory geography zone setup, you can import the territory zones into another Oracle Sales Cloud instance. For more information, refer to the Importing Setup Data listed in the related topics section.

---

**Note**

Ensure that you import your master reference geography data into the new Oracle Sales Cloud instance before you import the configuration package.

---

**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>Question</td>
<td>Answer</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>What is the structure of the geography types?</td>
<td>Create geography types with the following ranking structure:</td>
</tr>
<tr>
<td></td>
<td>1. County</td>
</tr>
<tr>
<td></td>
<td>2. Post Town</td>
</tr>
<tr>
<td>What is the geography hierarchy?</td>
<td>Create the following hierarchy:</td>
</tr>
<tr>
<td></td>
<td>1. Country of United Kingdom</td>
</tr>
<tr>
<td></td>
<td>2. County of Berkshire</td>
</tr>
<tr>
<td></td>
<td>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 Legal Jurisdictions and Authorities

Jurisdictions and Legal Authorities: Explained

You are required to register your legal entities with legal authorities in the jurisdictions where you conduct business. Register your legal entities as required by local business requirements or other relevant laws. For example, register your legal entities for tax reporting to report sales taxes or value added taxes.

Define jurisdictions and related legal authorities to support multiple legal entity registrations, which are used by Oracle Fusion Tax and Oracle Fusion Payroll. When you first create a legal entity, the Oracle Fusion Legal Entity Configurator automatically creates one legal reporting unit for that legal entity with a registration.

Jurisdictions: Explained

Jurisdiction is a physical territory such as a group of countries, country, state, county, or parish where a particular piece of legislation applies. French Labor
Law, Singapore Transactions Tax Law, and US Income Tax Laws are examples of particular legislation that apply to legal entities operating in different countries’ jurisdictions. Judicial authority may be exercised within a jurisdiction.

Types of jurisdictions are:

- Identifying Jurisdiction
- Income Tax Jurisdiction
- Transaction Tax Jurisdiction

**Identifying Jurisdiction**

For each legal entity, select an identifying jurisdiction. An identifying jurisdiction is your first jurisdiction you must register with to be allowed to do business in a country. If there is more than one jurisdiction that a legal entity needs to register with to commence business, select one as the identifying jurisdiction. Typically the identifying jurisdiction is the one you use to uniquely identify your legal entity.

Income tax jurisdictions and transaction tax jurisdictions do not represent the same jurisdiction. Although in some countries, the two jurisdictions are defined at the same geopolitical level, such as a country, and share the same legal authority, they are two distinct jurisdictions.

**Income Tax Jurisdiction**

Create income tax jurisdictions to properly report and remit income taxes to the legal authority. Income tax jurisdictions by law impose taxes on your financial income generated by all your entities within their jurisdiction. Income tax is a key source of funding that the government uses to fund its activities and serve the public.

**Transaction Tax Jurisdiction**

Create transaction tax jurisdictions through Oracle Fusion Tax in a separate business flow, because of the specific needs and complexities of various taxes. Tax jurisdictions and their respective rates are provided by suppliers and require periodic maintenance. Use transaction tax jurisdiction for legal reporting of sales and value added taxes.

**Legal Authorities: Explained**

A legal authority is a government or legal body that is charged with powers to make laws, levy and collect fees and taxes, and remit financial appropriations for a given jurisdiction.

For example, the Internal Revenue Service is the authority for enforcing income tax laws in United States. In some countries, such as India and Brazil, you are required to print legal authority information on your tax reports. Legal authorities are defined in the Oracle Fusion Legal Entity Configurator. Tax authorities are a subset of legal authorities and are defined using the same setup flow.
Legal authorities are not mandatory in Oracle Fusion Human Capital Management (HCM), but are recommended and are generally referenced on statutory reports.

Creating Legal Jurisdictions, Addresses and Authorities: Examples

Define legal jurisdictions and related legal authorities to support multiple legal entity registrations, which are used by Oracle Fusion Tax and Oracle Fusion Payroll.

Legal Jurisdictions
Create a legal jurisdiction by following these steps:

1. Navigate to the Manage Legal Jurisdictions page from the Setup and Maintenance work area by querying on the Manage Legal Jurisdictions task and selecting Go to Task.
2. Select Create.
3. Enter a unique Name, United States Income Tax.
4. Select a Territory, United States.
5. Select a Legislative Category, Income tax.
6. Select Identifying, Yes. Identifying indicates the first jurisdiction a legal entity must register with to do business in a country.
7. Enter a Start Date if desired. You can also add an End Date to indicate a date that the jurisdiction may no longer be used.
8. Select a Legal Entity Registration Code, EIN or TIN.
9. Select a Legal Reporting Unit Registration Code, Legal Reporting Unit Registration Number.
10. Optionally enter one or more Legal Functions.
11. Select Save and Close.

Legal Addresses for Legal Entities and Reporting Units
Create a legal address for legal entities and reporting units by following these steps:

1. Navigate to the Manage Legal Address page from the Setup and Maintenance work area by querying on the Manage Legal Address task and selecting Go to Task.
2. Select Create.
4. Enter Address Line 1, Oracle Parkway.
5. Optionally enter Address Line 2, and Address Line 3.
6. Enter or Select Zip Code, 94065.
7. Select Geography 94065 and Parent Geography Redwood Shores, San Mateo, CA.
9. Select **OK**.
10. Select **Save and Close**.

**Legal Authorities**

Create a legal authority by following these steps:

1. Navigate to the **Manage Legal Authorities** page from the **Setup and Maintenance** work area by querying on the **Manage Legal Authorities** task and selecting **Go to Task**.
2. Enter the **Name**, California Franchise Tax Board.
3. Enter the **Tax Authority Type**, Reporting.

---

**Note**

Create an address for the legal authority.

4. Select **Create**.
5. The **Site Number** is automatically assigned.
6. Optionally enter a **Mail Stop**.
7. Select **Country**, United States
8. Enter **Address Line 1**, 121 Spear Street, Suite 400.
9. Optionally enter **Address Line 2**, and **Address Line 3**.
10. Enter or Select **Zip Code**, 94105.
11. Select **Geography** 94105 and **Parent Geography** San Francisco, San Francisco, CA.
12. Select **OK**.
14. Optionally click the **One-Time Address** check box.
15. The **From Date** defaults to today’s date. Update if necessary.
16. Optionally enter a **To Date** to indicate the last day the address can be used.

---

**Note**

You can optionally enter **Address Purpose** details.

17. Select **Add Row**.
18. Select **Purpose**.
19. The **Purpose from Date** will default to today’s date.
20. Optionally enter a **Purpose to Date**.
21. Select **OK**.
22. Select **Save and Close**.
Creating Legal Entities, Registrations, and Reporting Units: Examples

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.

**Legal Entity**
From within an implementation project, create a legal entity by following these steps:

1. Navigate to an implementation project that contains the Define Legal Entities task list from the Setup and Maintenance work area.
2. Select Go to Task for the Define Legal Entities task list within the implementation project.

The following message appears:
You must first select a scope value to perform the task.
- Select and add an existing scope value to the implementation project.
- Create a new scope value and then add it to the implementation project.

3. Select Create New.
4. From the Manage Legal Entities page select Create.
5. Accept the default Country, United States.
6. Enter Name, InFusion USA West.
7. Enter Legal Entity Identifier, US0033.
8. Optionally enter Start Date. When the start date is blank the legal entity is effective from the creation date.
9. Optionally enter an End Date.
10. Optionally, if your legal entity should be registered to report payroll tax and social insurance, select the Payroll statutory unit check box.
11. Optionally, if your legal entity has employees, select the Legal employer check box.
12. Optionally, if this legal entity is not a payroll statutory unit, select an existing payroll statutory unit to report payroll tax and social instance on behalf of this legal entity.

Note
Enter the Registration Information.

14. Search for and select a **Legal Address**, 500 Oracle Parkway, Redwood Shores, CA 94065.

**Note**
The legal address must have been entered previously using the **Manage Legal Address** task.

15. Select **OK**.
16. Optionally enter a **Place of Registration**.
17. Enter the **EIN or TIN**.
18. Enter the **Legal Reporting Unit Registration Number**.
19. Select **Save and Close** to navigate back to the Manage Legal Entities page.
20. Select **Done** to return to your implementation project. An issue with the done button has been fixed in 11g Release 1 (11.1.4).
21. In the **Legal Entity** choice list in the implementation project (just below the implementation project name and code), click **Select and Add Legal Entity** to choose the legal entity that you just created, and set the scope for the remainder of your setup.
22. Search for and select your legal entity from the **Manage Legal Entities** page.
23. Select **Save and Close**.
   This sets the scope for your task list to the selected legal entity, as indicated in the **Legal Entity** choice list above the **Tasks and Task Lists** table.

**Legal Entity Registrations**
A legal entity registration with the same name as that of the legal entity will be created by default. To verify this, locate the **Manage Legal Entity Registrations** task and then select **Go to Task**. To create another registration for the legal entity follow these steps:

1. Navigate to your implementation project from the **Setup and Maintenance** work area. Verify that the parent **Legal Entity** scope value is set correctly.
2. Expand the **Define Legal Entities** task list within the implementation project.
3. Select **Manage Legal Entity Registrations Go to Task**.
4. Select **Create**.
5. Enter **Jurisdiction**.
6. Enter **Registered Address**.
7. Enter **Registered Name**.
8. Optionally enter Alternate Name, Registration Number, Place of Registration, Issuing Legal Authority, and Issuing Legal Authority Address, Start Date, and End Date.


Legal Reporting Unit

When a legal entity is created, a legal reporting unit with the same name as that of the entity is also automatically created. To create more legal reporting units or modify the settings follow these steps:

1. Navigate to your implementation project from the Setup and Maintenance work area. Verify that the parent Legal Entity scope value is set correctly.

2. Select Go to Task for the Define Legal Entities task list within the implementation project.

3. Select Create.

4. Enter Territory, United States.

5. Enter Name.

6. Optionally enter a Start Date.

Note
Enter Registration Information.

7. Search for and select Jurisdiction.

Note
Enter Main Legal Reporting Unit information.

8. Select the value Yes or No for the Main Legal Reporting Unit. Set value to yes only if you are creating a new main (primary) legal reporting unit.

9. Enter the Main Effective Start Date, 1/1/11.

10. Save and Close.

Define Legal Entities: Manage Legal Entity

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.

Define Legal Entities: Manage Legal Entity HCM Information

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 Manage Legal Entity HCM Information

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 Legal Entities: Manage Legal Entity Tax Profile

Party Tax Profiles: Explained

A tax profile is the body of information that relates to a party’s transaction tax activities. A tax profile can include main and default information, tax registration, tax exemptions, party fiscal classifications, tax reporting codes, configuration options, and service subscriptions.
Set up tax profiles for the following parties involved in your transactions:

- **First parties**: All legal entities, legal reporting units, and business units in your organization that have a transaction tax requirement.
- **Third parties**: Your customers and suppliers and their locations and banks.
- **Tax authorities**: Parties that administer tax rules and regulations.

**First Parties**

Set up tax profiles for your first party legal entities, legal reporting units, and business units.

First party legal entities identify your organization to the relevant legal authorities, for example, a national or international headquarters. Legal entities let you more accurately model your external relationships to legal authorities. The relationships between first party legal entities and the relevant tax authorities normally control the setup of the transaction taxes required by your business. Under most circumstances the tax setup is used and maintained based on the configuration of the legal entity. Enter the default information, party fiscal classifications, tax reporting codes, and configuration options for your legal entities. You can also specify if you are using the tax services of an external service provider for tax calculation.

First party legal reporting units identify each office, service center, warehouse and any other location within the organization that has a tax requirement. A legal reporting unit tax profile is automatically created for the headquarter legal entity. Set up additional legal reporting unit tax profiles for those needed for tax purposes. For legal reporting units, enter the default information, tax registrations, party fiscal classifications, and tax reporting codes. Also, define tax reporting details for your VAT and global tax reporting needs for tax registrations of tax regimes that allow this setup.

Business units organize your company data according to your internal accounting, financial monitoring, and reporting requirements. To help you manage the tax needs of your business units, you can use the business unit tax profile in either of two ways:

- Indicate that business unit tax setup is used and maintained based on the configuration of the associated legal entity at transaction time. The tax setup of the associated legal entity setup is either specific to the legal entity or shared across legal entities using the Global Configuration Owner setup.

- Indicate that tax setup is used and maintained by a specific business unit. Create configuration options for the business unit to indicate that the subscribed tax content is used for the transactions created for the business unit.

For business units that maintain their own setup, enter the default information, tax reporting codes, configuration options, and service providers as required.

**Third Parties**

Set up third party tax profiles for parties with the usage of customer, supplier, and their sites. Enter the default information, tax registrations, party fiscal
classifications, and reporting codes required for your third parties or third party sites. You can set up tax exemptions for your customers and customer sites.

Banks are also considered third parties. When a bank is created, the tax registration number specified on the bank record is added to the party tax profile record in Oracle Fusion Tax. You can not modify the party tax profile for a bank as it is view only. You can only modify the bank record itself.

**Note**

Setting up party tax profiles for third parties is not required. Taxes are still calculated on transactions for third parties that do not have tax profiles.

**Tax Authorities**

Set up a tax authority party tax profile using the Legal Authorities set up task. The tax authority party tax profile identifies a tax authority party as a collecting authority or a reporting authority or both. A collecting tax authority manages the administration of tax remittances. A reporting tax authority receives and processes all company transaction tax reports.

The collecting and reporting tax authorities appear in the corresponding list of values on all applicable Oracle Fusion Tax pages. All tax authorities are available in the list of values as an issuing tax authority.

**Specifying First Party Tax Profile Options: Points to Consider**

Set up first party tax profiles for all legal entities, legal reporting units, and business units in your organization that have a transaction tax requirements. How you set up your first parties can impact the tax calculation on your transactions.

The first party tax profile consists of:

- Defaults and controls: Applicable to legal entities and legal reporting units. Business units that use their own tax setup do not have defaults and controls.
- Tax registrations: Applicable to legal reporting units.
- Party fiscal classifications: Applicable to legal entities and legal reporting units.
- Tax reporting codes: Applicable to legal entities, legal reporting units, and business units who do not use the tax setup of the legal entity.
- Configuration options: Applicable to legal entities and business units who do not use the tax setup of the legal entity.
- Service subscriptions: Applicable to legal entities and business units who do not use the tax setup of the legal entity.

**Defaults and Controls**

The following table describes the defaults and controls available at the first party tax profile level:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rounding Level</td>
<td>Perform rounding operations on the:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Header</strong>: Applies rounding to calculated tax amounts once for each tax rate per invoice.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Line</strong>: Applies rounding to the calculated tax amount on each invoice line.</td>
</tr>
<tr>
<td>Rounding Rule</td>
<td>The rule that defines how the rounding should be performed on a value involved in a taxable transaction. For example, up to the next highest value, down to the next lowest value, or nearest.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>If you defined a rounding precedence hierarchy in the configuration owner tax option settings for the combination of configuration owner and event class, Oracle Fusion Tax considers the rounding details in the applicable tax profile.</td>
</tr>
<tr>
<td>Set Invoice Values as Tax Inclusive</td>
<td>This first party intends to send or receive invoices with invoice line amount inclusive of the tax amount.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>This option overrides the tax inclusive handling setting at the tax level, but not at the tax rate level.</td>
</tr>
</tbody>
</table>

**Tax Registrations**

You must set up a separate tax registration to represent each distinct registration requirement for a first party legal reporting unit. Oracle Fusion Tax uses tax registrations in tax determination and tax reporting. If your first party has more than one tax registration under the same tax regime, then the application considers the tax registration in the order: tax jurisdiction; tax; tax regime.

You must enable the **Use tax reporting configuration** option on the first party tax regime to allow entry of global tax reporting configuration details during tax registration setup for legal reporting units for these tax regimes.

**Party Fiscal Classifications**

If applicable, associate first party fiscal classification codes with this party. The party fiscal classification codes you enter become part of tax determination for invoices associated with this party. Specify start and end dates to control when these fiscal classifications are applicable for this party and transaction.

For legal entities, you can view the associated legal classifications that were assigned to the tax regime defined for this first party. The legal classifications are used in the tax determination process, similarly to the party fiscal classifications.
Tax Reporting Codes

Set up tax reporting types to capture additional tax information on transactions for your tax reports for your first parties. Depending on the tax reporting type code, you either enter or select a tax reporting code for this party. Specify start and end dates to control when these tax reporting codes are applicable.

Configuration Options

The legal entities and business units in your organization are each subject to specific sets of tax regulations as designated by the tax authorities where you do business. Use configuration options to associate legal entities and business units with their applicable tax regimes. You can set up tax configuration options when you create a tax regime or when you create a party tax profile. Both setup flows display and maintain the same party and tax regime definitions.

Service Subscriptions

Oracle Fusion Tax lets you use the tax services of external service providers for tax calculation of US Sales and Use Tax on Receivables transactions. The setup for provider services is called a service subscription. A service subscription applies to the transactions of one configuration option setup for a combination of tax regime and legal entity or business unit. Set up service subscriptions when you create a tax regime or when you create a party tax profile for a first party legal entity or business unit.

FAQs for Manage Legal Entity Tax Profile

When does a party tax profile get created for a legal entity?

The legal entity party tax profile is automatically created when a legal entity record is created. If a legal entity party tax profile record is not created, for example, when a legal entity is created through a back-end process, a legal entity party tax profile is created upon saving the tax regime when a legal entity is subscribed to or upon saving the configuration owner tax options when they are defined for the legal entity. Otherwise, create a party tax profile using the Create Legal Entity Tax Profile page. You can edit the tax profile that was automatically generated with the relevant tax information, but it is not required.

Define Legal Entities: Define Legal Reporting Units

Planning Legal Reporting Units: Points to Consider

Each of your legal entities has at least one legal reporting unit. Legal reporting units can also be referred to as establishments. You can define either domestic or foreign establishments. Define legal reporting units by physical location, such as
a sales office, or by logical unit, such as groups of employees subject to different reporting requirements. For example, define logical legal reporting units for both salaried and hourly paid employees.

Another example of logical reporting units is in the Human Capital Management (HCM) system where you use your legal reporting units to model your tax reporting units. A tax reporting unit is used to group workers for the purpose of tax reporting.

**Planning Legal Reporting Units**

Plan and define your legal reporting units at both the local and national levels if you operate within the administrative boundaries of a jurisdiction that is more granular than country. For example, your legal entity establishes operations in a country that requires reporting of employment and sales taxes locally as well as nationally. Therefore, you need more than one legally registered location to meet this legal entity's reporting requirements in each local area. Additionally, legal entities in Europe operate across national boundaries, and require you to set up legal reporting units for the purposes of local registration in each country. There can be multiple registrations associated with a legal reporting unit. However, there can be only one identifying registration, defined by the legal authority used for the legal entity or legal reporting unit, associated with the legal reporting unit.

**Define Chart of Accounts for Enterprise Structures: Manage Chart of Accounts Structures and Structure Instances**

**Chart of Accounts: Explained**

The chart of accounts is the underlying structure for organizing financial information and reporting. An entity records transactions with a set of codes representing balances by type, expenses by function, and other divisional or organizational codes that are important to its business.

A well-designed chart of accounts provides the following benefits:

- Effectively manages an organization’s financial business
- Supports the audit and control of financial transactions
- Provides flexibility for management reporting and analysis
- Anticipates growth and maintenance needs as organizational changes occur
- Facilitates an efficient data processing flow
- Allows for delegation of responsibility for cost control, profit attainment, and asset utilization
- Measures performance against corporate objectives by your managers

The chart of accounts facilitates aggregating data from different operations, from within an operation, and from different business flows, thus enabling the organization to report using consistent definitions to their stakeholders in compliance with legislative and corporate reporting standards and aiding in management decisions.
Best practices include starting the design from external and management reporting requirements and making decisions about data storage in the general ledger, including thick versus thin general ledger concepts.

**Thick Versus Thin General Ledger: Critical Choices**

Thick versus thin general ledger is standard terminology used to describe the amount of data populated and analysis performed in your general ledger. Thick and thin are the poles; most implementations are somewhere in between. Here are some variations to consider:

- A general ledger used in conjunction with an enterprise profitability management (EPM) product, which has data standardized from each operation, is designed as a thin general ledger. Use this variation if your solution is project based, and Oracle Fusion Projects is implemented. More detailed reporting can be obtained from the Projects system. In the thin general ledger, business units, divisions, and individual departments are not represented in the chart of accounts.

- A general ledger, with segments representing all aspects and capturing every detail of your business, with frequent posting, many values in each segment, and many segments, is called a thick general ledger. A thick general ledger is designed to serve as a repository of management data for a certain level of management. For example, a subsidiary’s general ledger is designed to provide the upper management enough data to supervise operations, such as daily sales, without invoice details or inventory without part number details.

- A primary ledger and a secondary ledger, where one is a thick general ledger and the other a thin general ledger, provides dual representation for reporting requirements that require more than one ledger.

**Thin General Ledger**

With a thin general ledger, you use the general ledger for internal control, statutory reporting, and tracking of asset ownership. You minimize the data stored in your general ledger. A thin general ledger has many of the following characteristics:

- Minimal chart of accounts
- Short list of cost centers
- Short list of natural accounts
  - Short list of cost accounts
  - Summary level asset and liability accounts
- Low number of optional segments
- Infrequent posting schedule

A thin general ledger has natural accounts at a statutory reporting level, for example, payroll expense, rent, property taxes, and utilities. It has cost centers...
at the functional expense level, such as Research and Development (R&D) or Selling, General, and Administrative (SG&A) expense lines, rather than at department or analytic levels. It omits business unit, division, and product detail.

One example of an industry that frequently uses a thin general ledger is retail. In a retail organization, the general ledger tracks overall sales numbers by region. A retail point of sales product tracks sales and inventory by store, product, supplier, markup, and other retail sales measures.

**Thick General Ledger**

With a thick general ledger, you use the general ledger as a detailed, analytic tool, performing analytic functions directly in the general ledger. Data is broken down by many reporting labels, and populated frequently from the subledgers. You maximize the data stored in the general ledger. A thick general ledger has many of the following characteristics:

- Maximum use of the chart of accounts
- Long list of natural accounts
- Long list of cost centers
- Long list of costing accounts
- Detailed asset and liability accounts
- Frequent posting schedule

In a thick general ledger, you obtain detail for cost of goods sold and inventory balances and track property plant and equipment at a granular level. Cost centers represent functional expenses, but also roll up to departmental or other expense analysis levels. Using product and location codes in optional segments can provide reporting by line of business. Posting daily, at the individual transaction level, can maximize the data stored in the general ledger.

One example of an industry that frequently uses a thick general ledger is electronic manufacturers. Detail on the revenue line is tagged by sales channel. Product is structured differently to provide detail on the cost of goods sold line, including your bill of materials costs. The general ledger is used to compare and contrast both revenue and cost of goods sold for margin analysis.

**Other Considerations**

Consider implementing a thick ledger if there are business requirements to do any of the following:

- Track entered currency balances at the level of an operational dimension or segment of your chart of accounts, such as by department or cost center
- Generate financial allocations at the level of an operational dimension or segment
- Report using multiple layered and versioned hierarchies of the operational dimension or segment from your general ledger

Consider implementing a thin ledger in addition to a thick ledger, if there are additional requirements for:
• Minimal disclosure to the authorities in addition to the requirements listed above. For example, in some European countries, fiscal authorities examine ledgers at the detailed account level.

• Fiscal only adjustments, allocations, and revaluations, which don’t impact the thick general ledger.

The important consideration in determining if a thick ledger is the primary or secondary ledger is your reporting needs. Other considerations include how the values for an operational dimension or segment are derived and the amount of resources used in reconciling your different ledgers. If values for the operational dimension are always entered by the user like other segments of the accounting flexfield, then a thick primary ledger is the better choice.

However, if values for the operational dimension or segment are automatically derived from other attributes on the transactions in your subledger accounting rules, rather than entered in the user interface, then use a thick secondary ledger. This decision affects the amount of:

• Storage and maintenance needed for both the general ledger and subledger accounting entries

• System resources required to perform additional posting

• In summary, you have:
  • Minimum demand on storage, maintenance, and system resources with the use of a thin ledger
  • Greater demand on storage, maintenance, and system resources with the use of a thick ledger
  • Greatest demand on storage, maintenance and system resources with the use of both thick and thin ledgers

Note

Generally speaking, there is a tradeoff between the volume of journals and balances created and maintained versus system resource demands. Actual performance depends on a wide range of factors including hardware and network considerations, transaction volume, and data retention policies.

Summary

The factors you need to consider in your decision to use a thick or thin general ledger for your organization, are your:

• Downstream EPM system and its capabilities

• Business intelligence system and its capabilities

• Subledger systems and their capabilities and characteristics, including heterogeneity

• General ledger reporting systems and their capabilities

• Maintenance required for the thick or thin distributions and record keeping

• Maintenance required to update value sets for the chart of accounts segments

• Preferences of the product that serves as a source of truth
• Level at which to report profitability including gross margin analysis
• Industry and business complexity

Chart of Accounts: How Its Components Fit Together

There are several important elements to the basic chart of accounts in Oracle Fusion Applications: a structure that defines the account values, segments, and their labels, and rules (security and validation). Account combinations link the values in the segments together and provide the accounting mechanism to capture financial transactions.

Chart of Accounts

The chart of accounts defines the number and attributes of various segments, including the order of segments, the width of segments, prompts, and segment labels, such as balancing, natural account, and cost center.

The chart of accounts further defines the combination of value sets associated with each segment of the chart of accounts, as well as the type, default value, additional conditions designating the source of the values using database tables, and the required and displayed properties for the segments.

Segments

A chart of accounts segment is a component of the account combination. Each segment has a value set attached to it to provide formatting and validation of the set of values used with that segment. The combination of segments creates the account combination used for recording and reporting financial transactions. Examples of segments that may be found in a chart of accounts are company, cost center, department, division, region, account, product, program, and location.
Value Sets and Values

The value sets define the attributes and values associated with a segment of the chart of accounts. You can think of a value set as a container for your values. You can set up your flexfield so that it automatically validates the segment values that you enter against a table of valid values. If you enter an invalid segment value, a list of valid values appears automatically so that you can select a valid value. You can assign a single value set to more than one segment, and you can share value sets across different flexfields.

Segment Labels

Segment labels identify certain segments in your chart of accounts and assign special functionality to those segments. Segment labels were referred to as flexfield qualifiers in Oracle E-Business Suite. Here are the segment labels that are available to use with the chart of accounts.

- **Balancing**: Ensures that all journals balance for each balancing segment value or combination of multiple balancing segment values to use in trial balance reporting. There are three balancing segment labels: primary, second, and third balancing. The primary balancing segment label is required.
- **Cost Center**: Facilitates grouping of natural accounts by functional cost types, accommodating tracking of specific business expenses across natural accounts. As cost centers combine expenses and headcount data into costs, they are useful for detailed analysis and reporting. Cost centers are optional, but required if you are accounting for depreciation, additions, and other transactions in Oracle Fusion Assets, and for storing expense approval limits in Oracle Fusion Expense Management.
- **Natural Account**: Determines the account type (asset, liability, expense, revenue, or equity) and other information specific to the segment value. The natural account segment label is required.
- **Management**: Optionally, denotes the segment that has management responsibility, such as the department, cost center, or line of business. Also can be attached to the same segment as one of the balancing segments to make legal entity reporting more granular.
- **Intercompany**: Optionally, assigns the segment to be used in intercompany balancing functionality.

Note

All segments have a segment qualifier that enables posting for each value. The predefined setting is Yes to post.

Account Combinations

An account combination is a completed code of segment values that uniquely identifies an account in the chart of accounts, for example 01-2900-500-123, might represent InFusion America (company)-Monitor Sales (division)-Revenue (account)-Air Filters (product).

Rules

The chart of accounts uses two different types of rules to control functionality.

- **Security rules**: Prohibit certain users from accessing specific segment values. For example, you can create a security rule that grants a user access only to his or her department.
• Cross-validation rules: Control the account combinations that can be created during data entry. For example, you may decide that sales cost centers 600 to 699 should enter amounts only to product sales accounts 4000 to 4999.

Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheets: Explained

Represent your enterprise structures in your chart of accounts, ledger, legal entities, and business unit configuration to track and report on your financial objectives and meet your reporting requirements. These components are the underlying structure for organizing financial information and reporting.

The chart of accounts within the ledger facilitates aggregating data from different operations, from within an operation, and from different business flows. This functionality enables you to report using consistent definitions to your stakeholders in compliance with legislative and corporate reporting standards and aids in management decisions.

Rapid implementation is a way to configure the Oracle Fusion Financial Enterprise and Financial Reporting Structures quickly using sheets in a workbook to upload lists of companies (legal entities), ledgers, business units, chart of account values, and other similar data. Once the sheets have been uploaded, the application creates your ledger, business unit, and other components. The following graphic shows the relationship of these components.

• Legal Entities: Identifies a recognized party with rights and responsibilities given by legislation, which has the right to own property and the responsibility to account for themselves.
• Chart of Accounts: Configures accounts consisting of components called segments that are used to record balances and organize your financial information and reporting.

• Segments: Contains a value set that provides formatting and validation of the set of values used with that segment. When combined, several segments create an account for recording your transactions and journal entries.

• Segment Labels: Identifies certain segments in your chart of accounts and assigns special functionality to those segments. The three required segment labels are:
  • Balancing Segment: Ensures that all journals balance for each balancing segment value or combination of multiple balancing segment values to use in financial processes and reporting. The three balancing segment labels are: primary, second, and third balancing. The primary balancing segment label is required.
  • Natural Account: Facilities processes in the General Ledger application, such as retained earnings posting. Determines the account type, which includes asset, liability, expense, revenue, or equity.
  • Cost Center: Facilitates grouping of natural accounts by functional cost types, accommodating tracking of specific business expenses across natural accounts.

• Ledger: Maintains the records and is a required component in your configuration. The Rapid implementation process:
  • Creates your ledger by combining your chart of accounts, calendar, and currency as well as other required options defined in the sheets.
  • Assigns a default for the fourth component, the subledger accounting method, used to group subledger journal entry rule sets together to define a consistent accounting treatment.
  • Creates a balances cube for each ledger with a unique chart of accounts and calendar. Each segment is created as a dimension in the balances cube.

• Business Units with Business Functions: Identifies where subledger transactions are posted and provides access to perform subledger business processes. Business units are assigned to a primary ledger, as well as a default legal entity, when configured and identify where subledger transactions are posted.

• Subledgers: Captures detailed transactional information, such as supplier invoices, customer payments, and asset acquisitions. Uses subledger accounting to transfer transactional balances to the ledger where they are posted.

Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheets: How They Are Processed

The Create Chart of Accounts, Ledger, Legal Entities, and Business Units rapid implementation process consists of four steps.
1. Enter the data into the sheets.
2. Upload the XML files generated from the sheets.
3. Run the deployment process to finalize the chart of accounts configuration.
4. Upload the XML files generated from the sheets for the rest of the configuration.

**Note**
On the Instruction sheet is a link to a completed sample data workbook.

**Process Overview**
Begin by downloading the Rapid Implementation for General Ledger workbook using the Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheet task on the Setup and Maintenance work area.

The following figure illustrates the Create Chart of Accounts, Ledger, Legal Entities, and Business Units process, what data is entered into each sheet of the workbook, and the components that the process creates.
Process

Enter Data

The Create Chart of Accounts, Ledger, Legal Entities, and Business Units workbook provides five sheets.

1. Instructions
2. Chart of Accounts, Calendar, and Ledger
3. Business Units
4. Companies and Legal Entities
5. Natural Accounts

Sheets used to enter other segment values and hierarchies for additional segments are created by entering the segments on the Chart of Accounts, Calendar, and Ledger sheet and then clicking the Add Segment Sheets button.

Instructions Sheet

Read the planning tips, loading process, best practices, and recommendations.

Chart of Accounts, Calendar, and Ledger Sheet

Enter your data to create your ledger, its components, chart of accounts, currency, and calendar, and set the required ledger options.

- **Ledger** name is the name of your primary ledger and often appears in report titles, so enter a printable name.
- **Ledger Currency** represents the currency that most of your transactions are entered.
- **Retained Earnings Account** is used when you open the first period of a new year. The application moves the total balances in your revenue and expense accounts to the Retained Earnings accounts by balancing segment.

Tip

When the data is uploaded, the Allow Dynamic Insertion option used to enable the generation of new account combinations dynamically instead of creating them manually is enabled by default. To prevent the creation of invalid accounts, you must define cross-validation rules. Define cross-validation rules before...
entering data or loading history. Cross-validation rules only prevent creation of new accounts, not disabling of preexisting accounts.

- **Enable Average Balances** is used to enable Average Balances functionality.

  The Average Balance feature provides organizations with the ability to track average and end-of-day balances, report average balance sheets, and create custom reports using both standard and average balances. Average balance processing is important for financial institutions, since average balance sheets are required, in addition to standard balance sheets, by many regulatory agencies. Many organizations also use average balances for internal management reporting and profitability analysis.

**Tip**

If you select No and uploaded the options, this region cannot be changed and does not display on the Specify Ledger Options page.

- **Fiscal Year Start Date** is the beginning date of your calendar for the ledger and cannot be changed once the ledger is saved.

**Important**

Select a period before the first period you plan to load history or perform translations to enable running translation. You cannot run translation in the first defined period of a ledger calendar.

- **Period Frequency** must be Monthly and is predefined.

**Note**

If you require a calendar other than monthly, such as 4-4-5 or weekly, define the calendar in the regular calendar page.

- **Adjusting Periods** add one or more periods that are used to enter closing, audit, or other adjustments in the General Ledger at quarter or year end. The entries are tracked in the adjusting period and not in your monthly activity.

- **Chart of Accounts** region is where you enter your segments, segment labels, short prompts, and display length data that is used to create your chart of accounts. Plan this data carefully, as you are defining the basic structure for your accounting and reporting.

- **Display Length** sets the segment size so select carefully and leave room for growth. For example, if you have 89 cost centers, enter 3 for the Display Length to allow for more than 100 cost centers in the future.

- **Add Segment Sheets** button to create sheets for additional segments. Only the Company and Natural Account segment sheets are provided.

**Note**
If you select an intercompany segment, you must complete at least one intercompany rule and check the Enable Intercompany Balancing option in the Specify Ledger Options task for the Balancing API to perform intercompany balancing.

**Business Units Sheet**

Enter the name of your business unit.

You can enter more than one business unit per ledger but it is not recommended.

![Business Units](image)

**Note**

Enter a list of your legal entities. Include their registration number and assigned parent or child value.

**Companies and Legal Entities Sheet**

You can create up to 9 levels of parent values to use to roll up your legal entities to meet corporate and local reporting requirements.

![Companies and Legal Entities](image)

**Natural Accounts Sheet**

Enter your account values that are used to record the type of balance.

![Natural Accounts](image)

- **Parent and Child Values** with **Descriptions** are used to build hierarchies. Hierarchies are used for chart of accounts mappings, revaluations, data access sets, cross validation rules, and segment value security rules. The balances cube and account hierarchies are also used for financial reporting, Smart View queries, and allocations.

- **Account Type** is used to identify the type of account, Asset, Liability, Revenue, Expense, or Owner’s Equity. Account types are used in year end close processes and to correctly categorize your account balances for reporting.

- **Financial Category** (optional) is used to identify groups of accounts for reporting with Oracle Fusion Transactional Business Intelligence.
Upload the Sheets and Run Deployment

Return to the Chart of Accounts, Calendar, and Ledger sheet after completing the other sheets complete the following steps:

1. (B) Generate Chart of Accounts File: The program generates an XML data file for the entered chart of accounts and hierarchies setup data. Save the file to a network or local drive.

2. (B) Generate Ledger, Legal Entity, and Business Units File: The program generates an XML data file for the entered ledger, legal entities, and business unit setup data. Save the file a network or local drive.

3. (N) Setup and Maintenance > Functional Setup Manager > Upload Chart of Accounts task. The Upload Enterprise Structures process is launched.

4. (B) Upload File.

5. (B) Browse. Select the first file you saved: ChartOfAccounts.xml

6. (B) Submit.

7. Verify that the process was completed without errors or warnings.

8. (N) Setup and Maintenance > Deploy Chart of Accounts task > (B) Deploy the Accounting Flexfield.

9. (I) Refresh until the green check mark appears and verifies that the deployment is successful.

10. (N) Setup and Maintenance > Upload Ledger, Legal Entities, and Business Units task. The Upload Enterprise Structures process is launched.
11. **(B) Upload File.**

12. **(B) Browse.** Select the second file you saved: `FinancialsCommonEntities.xml`

13. **(B) Submit.**

14. Verify that the process was completed without errors or warnings.

---

**Tip**

You cannot change the chart of accounts, accounting calendar, or currency for your ledger after the setup is created. Assign the data role that was automatically generated for the ledger to your users. Then open the first period to begin entering data.

---

**Creating One Chart of Accounts Structure with Many Instances: Example**

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created.

**Scenario**

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

**Analysis**

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created.

**Chart of Accounts Model**

The chart of accounts structure provides the general outline of the chart of accounts and determines the number of segments, the type, the length, and the label (qualifier) of each segment. This forms the foundation of the chart of accounts definition object.

For each chart of accounts structure, it is possible to associate one or more chart of accounts structure instances. Chart of accounts structure instances under the same structure share a common configuration with the same segments, in the
same order, and the same characteristics. Using one chart of accounts structure with multiple instances simplifies your accounting and reporting.

At the chart of accounts structure instance level, each segment is associated with a value set that conforms to the characteristic of that segment. For example, you assign a value set with the same segment type and length to each segment. You are using hierarchies with your chart of accounts segments. Each structure instance segment is assigned a tree code to indicate the source of the hierarchy information for the associated value set. The same value set can be used multiple times within the same or across different chart of accounts instances within the same structure or in different structures. This functionality reduces your segment value creation and maintenance across your charts of accounts.

The collective assignment of value sets to each of the segments forms one chart of accounts instance. At the chart of accounts structure instance level, you can select to enable dynamic insertion. Dynamic insertion allows the creation of account code combinations automatically the first time your users enter that new account combination. The alternative is to create them manually. By deciding to enable dynamic insertion, you save data entry time and prevent delays caused by the manual creation of new code combinations. Well defined cross validation rules help prevent the creation of inappropriate account code combinations.

Perform deployment after a new chart of accounts structure and structure instances are defined or any of their modifiable attributes are updated. Deployment validates and regenerates the necessary objects to enable your charts of accounts and chart of accounts structure instances. By unifying and standardizing your organization’s chart of accounts, you are positioned to take full advantage of future functionality in Oracle Fusion General Ledger.

In summary, you are recommending to your company to unify the organization’s chart of accounts in a single chart of accounts structure based on chart of accounts commonalities across ledgers. You have also decided to use the chart of accounts structure instance construct to serve different accounting and reporting requirements by using value sets specific to each of your entities.

**Creating Chart of Accounts Structure and Instances: Examples**

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created. A chart of accounts structure defines the key attributes for your chart of accounts, such as the number of segments, the segment sequences, the segment names, segment prompts, segment labels, for example natural account and primary balancing, and default value sets.

The chart of accounts instance is exposed in the user interfaces and processes. By default, a chart of accounts instance inherits all the attributes of the chart of accounts structure, meaning that all instances of the same structure share a common shape and have the same segments in the same order. However, at the chart of accounts instance level, you can override the default value set assignments for your segments and assign a unique account hierarchy that determines the parent and child relationships between the value set values. At the chart of accounts instance level, determine if allow dynamic insertion is enabled to generate new account combinations dynamically instead of creating them manually.
Chart of Account Structure

You are creating a chart of accounts structure as you setup your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

1. Navigate to the Manage Chart of Accounts page from the Functional Setup Manager by querying on Manage Chart of Accounts and clicking the Go To Task.

2. Select General Ledger from the Module list of values and click Search.

3. Click Manage Structures to open the Manage Key Flexfield Structures page.

4. Select the General Ledger row and click the Create to open the Create Key Flexfield Structure page.

5. Enter a unique Structure Code, INFUSION_AM_COA_STRUCTURE, and Name, InFusion America COA Structure. Provide an optional Description, InFusion America Inc. Chart of Accounts Structure.

6. Select the - Delimiter to visually separate your segment values.

7. Click Save.

8. To create a new segment, click the Create to open the Create Key Flexfield Segment page.

   a. Enter the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment Code</td>
<td>INFUSION_AM_CO</td>
</tr>
<tr>
<td>Name</td>
<td>InFusion America Company</td>
</tr>
<tr>
<td>Description</td>
<td>InFusion America Inc. Company</td>
</tr>
<tr>
<td>Sequence Number</td>
<td>1</td>
</tr>
<tr>
<td>Prompt</td>
<td>Company</td>
</tr>
<tr>
<td>Short Prompt</td>
<td>CO</td>
</tr>
<tr>
<td>Display Width</td>
<td>2</td>
</tr>
<tr>
<td>Column Name</td>
<td>Segment1</td>
</tr>
<tr>
<td>Default Value Set Code</td>
<td>INFUSION_AM_COMPANY</td>
</tr>
</tbody>
</table>

   b. Select a segment label, Primary Balancing Segment, to indicate its purpose within your chart of accounts.

   __Note__

   Two segment labels are required: primary balancing segment and natural account segment. These labels are not used with each other or with other labels in a specific segment.

   c. Click Save and Close.

   d. Click Done.
e. Define additional segments following the same process.

**Chart of Account Instance**

You are creating a chart of accounts instance as you setup your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

1. Navigate to the Manage Chart of Accounts page from the Functional Setup Manager by querying on Manage Chart of Accounts and clicking the Go To Task.
2. Select General Ledger from the Module list of values and click Search.
3. Select the General Ledger row and click Manage Structure Instances to open the Manage Key Flexfield Structure Instance page.
4. Click the Create icon to open the Create Key Flexfield Structure Instance page.
5. Enter a unique Structure Instance Code, INFUSION_AM_COA_INSTANCE, and Name, InFusion America COA Instance. Provide an optional Description, InFusion America Inc. Chart of Accounts Structure Instance.
6. Select Dynamic combination creation allowed to indicate that you want to dynamically generate account combinations.
7. Associate your instance with your Structure Name, InFusion America Structure.

**Note**

By default, an instance inherits the key attributes of the associated structure. Some attributes, such as the value set assigned to each the segment, can be modified.

8. Click Save.
9. Optionally, select the segment row and click Edit to modify instance segments.
10. Check Required, Displayed, and BI enabled check boxes.

**Note**

Check the Required and Displayed options for all segments including those intended for future use. The recommended best practice is to define one segment for future use and set a default value. This ensures room for expansion in your chart of accounts and that the extra segment is populated in the account combinations.

Check the BI (Business Intelligence) enabled option to use key flexfield segments in Oracle Fusion Transactional Business Intelligence. The business intelligence check box is only valid when enabled on segments with segment labels. The second step is to populate the BI Object Name field for each of the segment labels in the Manage Segment Label page opened from the Manage Key Flexfields page.
11. Click **OK**.

12. Click **Save and Close**.

13. Define additional instances following the same process.

---

**Note**

Alternatively, proceed directly with creating your value set values by selecting the corresponding **Value Set Code** in the Segment Instances table.

---

14. Click **Done**.

15. Click **Deploy Flexfield**.

16. Click **OK**.

---

**Balancing Segments: Explained**

Balancing segments ensure that all journals balance for each balancing segment value or combination of multiple balancing segment values. You can secure access to your primary balancing segment values only with data access sets. The general ledger application automatically calculates and creates balancing lines as required in journal entries. For example, recognizing an entity’s receivable and the other entity’s payable. There are three balancing segment labels: primary, second, and third balancing. The primary balancing segment label is required.

By enabling multiple balancing segments for your chart of accounts, it is possible to produce financial statements for each unique combination of segment values across not only one, but two or even three qualified balancing segments. This ability provides you greater insights into your operations as it affords you visibility along the critical fiscal dimensions you use to plan, monitor, and measure your financial performance.

The following explains processes that use balancing segments.

- **Intercompany balancing**: Adds lines to unbalanced journals using intercompany rules.

- **Opening first period of the new accounting year**: Calculates retained earnings amounts at the level of granularity that totals revenue and expense account balances for multiple balancing segment value combinations. This applies to standard and average balances.

- **Importing journals**: Adds lines using the suspense account on unbalanced journals.

- **Posting journals**: Adds additional lines to unbalanced journals for the following enabled account types:
  - Suspense
  - Rounding
  - Net income
  - Retained earnings
• Cumulative translation adjustments from replication of revaluation journals to reporting currencies and for multiple reporting currency account type specific conversion

• Posting prior period journals: Calculates any income statement impact and posts to the appropriate retained earnings account.

• Translating balances: Supports multiple balancing segments for the following accounts:
  - Retained earnings: Calculated translated retained earnings are post to the retained earnings accounts by balancing segment. Retained earnings represents the summing of the translated revenue and expense accounts across multiple balancing segment values.
  - Cumulative translation adjustment: Amounts posted by balancing segment to these accounts represents currency fluctuation differences between ranges of accounts which use different rate types. For example, period end rates are used for asset and liability accounts and historical rates for equity accounts.

• Revaluing Balances: Supports multiple balancing segments when calculating gain or loss accounts.

• Creating Opening Balances: Initializes reporting currency balances by converting from the total primary currency. Any difference in the reporting currency amounts is offset by populating retained earnings accounts.

• Closing year end: Supports multiple balancing segments when calculating the income statement offset and closing account in the closing journals.

### Multiple Balancing Segments: Points to Consider

Oracle Fusion General Ledger supports tracking financial results at a finer level of granularity than a single balancing segment. In addition to the required primary balancing segment for the chart of accounts, which is typically associated with the company dimension of a business organization, two additional segments of the chart of accounts can be optionally qualified as the second and third balancing segments respectively. Possible chart of accounts segments that can be tagged as these additional balancing segments include cost center or department, additional aspects of a business commonly used in measuring financial results.

There are several points to consider in using multiple balancing segments:

- Journal entry processing
- Implementation timing
- Change options
- Migration adjustments

### Journal Entry Processing

Multiple balancing segments ensure that account balances come from journal entries where the debits equal the credits, and thus, the financial reports are
properly generated for each unique instance of account value combinations across the balancing segments. Consider this option carefully as it provides more granular reporting but requires more processing resources.

**Implementation Timing**

When considering implementing the optional second and third balancing segments, keep in mind that these chart of accounts segment labels are set from the beginning of time and are actively used by your ledgers. This is important to ensure that balances are immediately maintained in accordance with the necessary balancing actions to produce consistent financial reporting for the desired business dimensions. Multiple balancing segment ledgers that are not maintained from the beginning of time require extensive manual balance adjustments to catch up and realign the balances in accordance with the multiple balancing segments.

---

**Note**

Do not set a segment already qualified as a natural account or intercompany segment as any of the three balancing segments. Validations are not performed when segment labels are assigned, so verify that all are assigned correctly before using your chart of accounts.

---

**Change Options**

Once a segment has been enabled and designated as a balancing segment, you must not change the segment. Do not disable the segment or remove the segment labels. These settings must be consistently maintained throughout the life of the chart of accounts to control the accuracy and integrity of the financial data.

---

**Migration Adjustments**

For charts of accounts migrated from Oracle E-Business Suite to Oracle Fusion General Ledger that use a segment with the secondary balance tracking segment qualifier, steps must be taken to ensure the proper transition to the second and third balancing segments. The required adjustments are extensive.

For ledgers associated with a migrated chart of accounts, its balances must be adjusted manually to be consistent with the second and third balancing segments as though these segment labels have been in place since the beginning of entries for these ledgers. This requires recomputing and updating of the following processes to reflect the correct balancing for each unique combination of segment values across the additional second and third balancing segments.

- Intercompany balancing
- Suspense posting
- Rounding imbalance adjustments on posting
- Entered currency balancing
- Revaluation gains or losses
- Retained earnings calculations at the opening of each new fiscal year
- Cumulative translation adjustments during translation

---

**Note**
All previously translated balances must also be purged, and new translations run to properly account for translated retained earnings and cumulative translation adjustments with the correct level of balancing.

**Using Multiple Balancing Segments: Example**

This simple example illustrates balancing along two balancing segments for a simple chart of accounts with three segments.

**Scenario**

Your company has a chart of accounts with two balancing segments and three segments, qualified as follows:

- Company: Primary balancing segment
- Cost Center: Second balancing segment
- Account: Natural account segment

The following multiple company and cost center journal has been entered to transfer advertising and phone expense from Company 1, Cost Center A to Company 2, Cost Center B.

<table>
<thead>
<tr>
<th>Account</th>
<th>Debit</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company 1-Cost Center A-Advertising Expense Account</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>Company 2-Cost Center B-Advertising Expense Account</td>
<td></td>
<td>600</td>
</tr>
<tr>
<td>Company 1-Cost Center A-Phone Expense Account</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>Company 2-Cost Center B-Phone Expense Account</td>
<td></td>
<td>800</td>
</tr>
</tbody>
</table>

During the posting process, the last four lines are created to balance the entry across the primary and second balancing segments, company and cost center.

<table>
<thead>
<tr>
<th>Account</th>
<th>Debit</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company 1-Cost Center A-Advertising Expense Account</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>Company 2-Cost Center B-Advertising Expense Account</td>
<td></td>
<td>600</td>
</tr>
<tr>
<td>Company 1-Cost Center A-Phone Expense Account</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>Company 2-Cost Center B-Phone Expense Account</td>
<td></td>
<td>800</td>
</tr>
<tr>
<td>Company 1-Cost Center A-Balancing Account</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>Company 2-Cost Center B-Balancing Account</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>Company 1-Cost Center A-Balancing Account</td>
<td></td>
<td>800</td>
</tr>
</tbody>
</table>

Common Applications Configuration: Define Enterprise Structures for Procurement 5-107
FAQs for Manage Charts of Accounts Structures and Structure Instances

How can I use future accounting segments?

To plan for future growth in the business organization that requires additional segments in the chart of accounts, extra segments can be added to the chart of accounts structure during your original implementation. Since all segments of the chart are required and have to be enabled, these unused segments can be assigned value sets that have a single value in the chart of accounts structure instance. This value is set as a default for that segment so that the extra segments are automatically populated when an account code combination is used.

Define Chart of Accounts for Enterprise Structures: Manage Chart of Accounts Value Sets and Value Set Values

Chart of Accounts Values Sets: Critical Choices

A value set is the collection of account values that are associated with a segment of a chart of accounts structure instance. When creating values sets, consider the following critical choices:

- Module Designation
- Validation Type
- Format Assignments
- Security Rules
- Values Definition

Module Designation

The module designation is used to tag value sets in Oracle Fusion Applications and sets the value sets apart during upgrades and other processes. Chart of accounts value sets upgraded from Oracle E-Business Suite Release 12 generically bear the module value of Oracle Fusion Middleware. When creating new value sets for a chart of accounts, the module can be specified as Oracle Fusion General Ledger to distinctly identify its intended use in an accounting flexfield, basically a chart of accounts.

Validation Type

Assign one of the following validation types to chart of accounts value sets:

- **Independent**: The values are independently selected when filling out the segment in the account combination.
• **Table Validated**: The values are stored in an external table to facilitate maintenance and sharing of the reference data.

**Format Assignments**

Value sets for chart of accounts must use the **Value Data Type** of **Character**. The **Value Subtype** is set to **Text**. These two settings support values that are both numbers and characters, which are typical in natural account segment values. Set the maximum length of the value set to correspond to the length of the chart of accounts segment to which it is assigned. Best practices recommend restricting values to **Upper Case Only** or **Numeric** values that are zero filled by default.

**Security Rules**

If flexfield data security rules are to be applied to the chart of accounts segment associated with the value set, the **Enable Security** check box must be checked for the assigned value set. In addition, assign a data security resource name to enable creation of a data security object automatically for the value set. The data security object is used in the definition of flexfield data security rules.

**Value Definition**

Once these basic characteristics are defined for the value set, values can be added to the set in the Manage Values page.

- Set the values to conform to the value set length and type.
- Enter the value, its description, and its attributes including the **Enable** check box, **Start Date**, and **End Date**.
- Assign the following attributes: **Parent** or **Summary** check box, **Posting is allowed**, and **Budgeting is allowed**.

**Note**

If the value set is used with a natural account segment, the value also requires you set the **Natural Account Type**, with one of the following values: **Asset**, **Liability**, **Equity**, **Revenue** or **Expense**. Other attributes used are **Third Party Control Account**, **Reconciliation** indicator, and **Financial Category** used with Oracle Transaction Business Intelligence reporting.

Oracle Fusion General Ledger best practice is to define the values for the value set after the value set is assigned to a chart of accounts structure instance. Otherwise you are not able to define the mandatory value attributes, such as summary flag, posting allowed, and account type for natural account segment. The attributes must be added after the value set is assigned to a chart of accounts structure instance.

**Creating a Value Set for Your Chart of Accounts: Example**

Create your value sets before creating your chart of accounts. A value set can be shared by different charts of accounts or across different segments of the same chart of accounts.
Scenario

You are creating a company value set to be used in your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

1. Navigate to the Manage Chart of Accounts Value Sets task from within your implementation project and click the Go to Task.
2. Click the Create icon on the toolbar of the Search Results table. The Create Value Set page opens.
3. Enter a unique Value Set Code, InFusion America Company, and an optional Description, Company values for InFusion America Inc.
4. Select General Ledger from the list in the Module field.
5. Select Independent as Validation Type.
6. Select Character as the Validation Data Type.
7. Click Save and Close.

Configuring Chart of Account Segment for Business Intelligence: Explained

To map the Oracle Fusion General Ledger Accounting Flexfield in Oracle Transaction Business Intelligence (BI) Repository file (RPD) for Oracle Fusion Financials, populate values in the Manage Key Flexfields user interface. These values enable the Chart of Accounts segments for Oracle Fusion Transactional BI and provide the mapping with BI Object names that are used as dimension for each of the Chart of Accounts segments.

Check each of the Chart of Accounts segments' BI enabled check box on all segments that you intend to map in the RPD by performing the following steps:

1. From your implementation project or the Setup and Maintenance page, query for Manage Key Flexfields and select the Go to Task.
2. Enter GL# in the Key Flexfield Code field.
3. Click Search button.
4. Click on Manage Structure Instances button.
5. Click the Search button.
6. Click on the desired chart of accounts and Edit icon.
7. Click on the desired segment and the Edit icon.
8. Edit each of the segments by checking the BI enabled check box.
9. Click on Save button. This should be done for all segments in every Chart of Accounts Structure Instance that you intend to be mapped in RPD.
10. Click the Save and Close button and the Done button.

Populate the BI Object Name for each of the Segment Labels. This name is the logical table name in the RPD which would be used as the dimension for the corresponding segment. Perform the following steps:

1. From your implementation project or the Setup and Maintenance page, query for Manage Key Flexfields and select the Go to Task.
2. Enter GL# in the **Key Flexfield Code** field.
3. Query for GL# as **Key Flexfield Code** in **Manage Key Flexfields** page.
4. Click **Search** button.
5. Chose **Actions** menu and click on **Manage Segment Labels**
6. Populate the **BI Object Name** for all the segment labels that are need to be mapped in the RPD.

<table>
<thead>
<tr>
<th>Segment Label Code</th>
<th>BI Object Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA_COST_CTR</td>
<td>Dim - Cost Center</td>
</tr>
<tr>
<td>GL_BALANCING</td>
<td>Dim - Balancing Segment</td>
</tr>
<tr>
<td>GL_ACCOUNT</td>
<td>Dim - Natural Account Segment</td>
</tr>
</tbody>
</table>

7. Click the **Save** button.

**Note**
For all the non qualified segment labels, the **BI Object Name** should be populated with one of the following:

- Dim - GL Segment1
- Dim - GL Segment2
- Dim - GL Segment3
- Dim - GL Segment4
- Dim - GL Segment5
- Dim - GL Segment6
- Dim - GL Segment7
- Dim - GL Segment8
- Dim - GL Segment9
- Dim - GL Segment10

Deploy the flexfield using the **Deploy Flexfield** button from **Manage Key Flexfields** page.

**Define Chart of Accounts for Enterprise Structures: Manage Accounting Calendars**

**Defining Accounting Calendars: Critical Choices**

Define an accounting calendar to create your accounting year and the periods it contains. Specify common calendar options that the application uses to automatically generate a calendar with its periods. Specifying all the options makes defining a correct calendar easier and more intuitive with fewer errors. The choices you make when specifying the following options are critical, because it is difficult to change your accounting calendar after a period status is set to open or future enterable.
• Start Date
• Period Frequency
• Adjusting Period Frequency
• Period Name Format

Note
In Oracle Fusion, the common calendar types, monthly, weekly, 4-4-5, 4-5-4, 5-4-4, 4-week, quarterly, and yearly, are automatically generated. This functionality makes it easier to create and maintain accounting calendars. By using the period frequency option, you no longer have to go through the tedious task of defining each period manually.

Start Date

If you plan to run translation, specify a calendar start date that is a full year before the start date of the year of the first translation period for your ledger. Translation cannot be run in the first period of a calendar. Consider how many years of history you are going to load from your previous system and back up the start date for those years plus one more. You cannot add previous years once the first calendar period has been opened.

Period Frequency

Use period frequency to set the interval for each subsequent period to occur, for example, monthly, quarterly, or yearly. If you select the period frequency of Other, by default, the application generates the period names, year, and quarter number. You specify the start and end dates. You must manually enter the period information. For example, select the period frequency of Other and enter 52 as the number of periods when you want to define a weekly calendar. For manually entered calendars, when you click the Add Year button, the application creates a blank year. Then, you must manually enter the periods for the new year. The online validation helps prevent erroneous entries.

If the year has been defined and validated, use the Add Year button to add the next year quickly. Accept or change the new rows as required. For example, with the Other frequency type calendar, dates may differ from what the application generates.

Note
In Oracle Fusion applications a calendar can only have one period frequency and period type. Therefore, if you have an existing calendar with more than one period type associated with it, during the upgrade from Oracle E-Business Suite, separate calendars are created based on each calendar name and period type combination.

Adjusting Period Frequency

Use the adjusting period frequency to control when the application creates adjusting periods. For example, some of the frequencies you select add one adjusting period at year end, two at year end, or one at the end of each quarter. The default is None which adds no adjusting periods. If you select the frequency
of Other, the Number of Adjusting Periods field is displayed. Enter the number of desired adjusting periods and then, manually define them.

**Period Name Format Region**

The User-Defined Prefix field in the Period Name Format region is an optional feature that allows you to enter your own prefix. For example, define a weekly calendar and then enter a prefix of Week, - as the separator, and the period name format of Period numberYY fiscal year. The application creates the names of Week1-11, Week2-11, through Week52-11. The options for the Format field are predefined values. The list of values is filtered based on the selected separator and only displays the options that match the selected separator.

The year displayed in the period names is based on the selected period name format and the dates the period covers or if the period crosses years, on the year of the start date of the period. For example, April 10, 2010 to May 9, 2010 has the period name of Apr-10 and December 10, 2010 to January 9, 2011 has the name of Dec-10. If period frequency is Other, then the period format region is hidden. The application generates a temporary period name for calendars with period frequency of Other, using a fixed format of Period numberYY. You can override this format with your own customized period names.

**Note**

For an accounting calendar that is associated with a ledger, changing period names or adding a year updates the accounting period dimension in the balances cubes.

---

**Calendar Validation: How It Works with the Accounting Calendar**

Calendar validation is automatic and prevents serious problems when you begin using the calendar. Once you set a calendar period status to open or future enterable, you cannot edit the period.

**Settings That Affect Calendar Validation**

The calendar validation runs automatically when you save the calendar.

**How the Calendar Is Validated**

The following table lists the validation checks performed when the accounting calendar is saved.

<table>
<thead>
<tr>
<th>Validation Performed</th>
<th>Example of Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique period number</td>
<td>2 assigned for two periods</td>
</tr>
<tr>
<td>Unique period name</td>
<td>Jan-11 entered twice</td>
</tr>
<tr>
<td>Period number beyond the maximum number of periods per year</td>
<td>13 for a 12 period calendar with no adjusting periods</td>
</tr>
<tr>
<td>Entered period name contains spaces</td>
<td>Jan 11</td>
</tr>
<tr>
<td>Single or double quotes in the period name</td>
<td>Jan '11</td>
</tr>
<tr>
<td>Nonadjusting periods with overlapping dates</td>
<td>01-Jan-2011 to 31-Jan-2011 and 30-Jan-2011 to 28-Feb-2011</td>
</tr>
</tbody>
</table>
Period date gaps | 01-Jan-2011 to 28-Jan-2011 and 31-Jan-2011 to 28-Feb-2011
Missing period numbers | Periods 1 through 6 defined for a twelve month calendar
Period number gaps | 1, 3, 5
Period numbers not in sequential order by date | Period 1 covers 01-Jan-2011 to 31-Jan-2011 and period 2 covers 01-Mar-2011 to 31-Mar-2011, and period 3 covers 01-Feb-2011 to 28-Feb-2011.
Quarter number gaps | 1, 3, 4
Quarters not in sequential order by period | 1, 3, 2, 4
Period start or end dates more than one year before or after the fiscal year | July 1, 2010 in a 2012 calendar

FAQs for Manage Accounting Calendars

How can I identify errors in my accounting calendar?

Oracle Fusion General Ledger identifies erroneous entries online as you enter a new calendar or change data on an existing calendar. The application also automatically validates the data when you save the calendar.

What's the difference between calendar and fiscal period naming?

The period naming format determines the year that is appended to the prefix for each period in the calendar. For the example, your accounting year has a set of twelve accounting period with a start date of September 1, 2011 and the end date is August 31, 2012, with each period’s date range following the natural calendar month date range.

Calendar period naming format: Select the calendar period format to append the period’s start date’s year to the prefix. For the period covering September 1, 2011 to December 31, 2011, then 2011 or just 11, depending on the period format selected, is appended to each period’s name. For the remaining periods covering January 1, 2012 to August 31, 2012, then 2012 or 12, is appended to each period’s name.

Fiscal period naming format: Select the fiscal period format to always append the period’s year assignment to the prefix. If the accounting periods in the set of twelve are all assigned the year of 2012, then 2012 or just 12, depending on the period format selected, is appended to the period name of all 12 periods.

When do I update an existing calendar?

Update an existing calendar before the new periods are needed as future periods, based on the future period setting in your accounting configuration. If a complete year has been defined and validated, use the Add Year button to add the next year quickly. Accept or change the new rows as required. For
example, with the Other frequency type calendar, dates may differ from what the application generates.

**What happens if I upgrade my calendar from Oracle E-Business Suite Release 12?**

The migration script assigns a period frequency that most closely matches your Oracle E-Business Suite Release 12 calendar. When you use the Oracle Fusion applications Add Year functionality for the first time, you have an opportunity to review and change the period frequency. The Calendar Options page opens only for calendars upgraded from Release 12 to allow one time modification.

Make your changes to the period frequency, adjusting period frequency, and period name format, including the prefix and separator, as needed. Changes can not conflict with the existing upgraded calendar definition. Update the calendar name and description in the calendar header, as needed, for all calendars.

Period details for a new year will be generated automatically based on the latest calendar options. You can also manually update the calendar. The modified calendar options affect future years only.

**Define Accounting Configurations of Enterprise Structures: Manage Primary or Secondary Ledgers**

**Accounting Configuration Offerings: Overview**

The Setup and Maintenance work area in the Oracle Fusion Applications is used to manage the configuration of legal entities, ledgers, and reporting currencies that comprise your accounting configuration. To create a new legal entity or ledger, your implementation consultant or system administrator must create an implementation project. This implementation project can be populated by either adding a financials related offering or one or more task lists.

**Note**

Setup tasks that are not related to the ledger or legal entity specific setup tasks can be invoked from either an implementation project or launched directly from the Setup and Maintenance work area.

There are two offerings predefined for financial implementations.

- The Oracle Fusion Accounting Hub offering is used to add the Oracle Fusion General Ledger and Oracle Fusion Subledger Accounting application features to an existing enterprise resource planning (ERP) system to enhance the current reporting and analysis.

- The Oracle Fusion Financials offering, which includes the Oracle Fusion General Ledger and Oracle Fusion Subledger Accounting application features, as well as at least one of the subledger financial applications.

When adding an offering to an implementation project, implementation consultants can customize the tasks displayed by adding additional tasks to the implementation project.
Ledgers and Subledgers: Explained

Oracle Fusion Applications reflect the traditional segregation between the general ledger and associated subledgers. Detailed transactional information is captured in the subledgers and periodically imported and posted in summary or detail to the ledger.

A ledger determines the currency, chart of accounts, accounting calendar, ledger processing options, and accounting method for its associated subledgers. Each accounting setup requires a primary ledger and optionally, one or more secondary ledgers and reporting currencies. Reporting currencies are associated with either a primary or secondary ledger.

The number of ledgers and subledgers is unlimited and determined by your business structure and reporting requirements.

Single Ledger

If your subsidiaries all share the same ledger with the parent company or they share the same chart of accounts and calendar, and all reside on the same applications instance, you can consolidate financial results in Oracle Fusion General Ledger in a single ledger. Use Oracle Fusion Financial Reporting functionality to produce individual entity reports by balancing segments. General Ledger has three balancing segments that can be combined to provide detailed reporting for each legal entity and then rolled up to provide consolidated financial statements.

Multiple Ledgers

Accounting operations using multiple ledgers can include single or multiple applications instances. You need multiple ledgers if one of the following is true:

- You have companies that require different account structures to record information about transactions and balances. For example, one company may require a six-segment account, while another needs only a three-segment account structure.
- You have companies that use different accounting calendars. For example, although companies may share fiscal year calendars, your retail operations require a weekly calendar, and a monthly calendar is required for your corporate headquarters.
- You have companies that require different functional currencies. Consider the business activities and reporting requirements of each company. If you must present financial statements in another country and currency, consider the accounting principles to which you must adhere.

Subledgers

Oracle Fusion Subledgers capture detailed transactional information, such as supplier invoices, customer payments, and asset acquisitions. Oracle Fusion Subledger Accounting is an open and flexible application that defines the accounting rules, generates detailed journal entries for these subledger
transactions, and posts these entries to the general ledger with flexible summarization options to provide a clear audit trail.

**Ledgers: Points to Consider**

Companies account for themselves in primary ledgers, and, if necessary, secondary ledgers and reporting currencies. Your transactions from your subledgers are posted to your primary ledgers and possibly, secondary ledgers or reporting currencies. Local and corporate compliance can be achieved through an optional secondary ledger, providing an alternate accounting method, or in some cases, a different chart of accounts. Your subsidiary’s primary and secondary ledgers can both be maintained in your local currency, and you can convert your local currency to your parent's ledger currency to report your consolidated financial results using reporting currencies or translation.

**Primary Ledgers**

A primary ledger is the main record-keeping ledger. Like any other ledger, a primary ledger records transactional balances by using a chart of accounts with a consistent calendar and currency, and accounting rules implemented in an accounting method. The primary ledger is closely associated with the subledger transactions and provides context and accounting for them.

To determine the number of primary ledgers, your enterprise structure analysis must begin with your financial, legal, and management reporting requirements. For example, if your company has separate subsidiaries in several countries worldwide, enable reporting for each country’s legal authorities by creating multiple primary ledgers that represent each country with the local currency, chart of accounts, calendar, and accounting method. Use reporting currencies linked to your country specific primary ledgers to report to your parent company from your foreign subsidiaries. Other considerations, such as corporate year end, ownership percentages, and local government regulations and taxation, also affect the number of primary ledgers required.

**Secondary Ledgers**

A secondary ledger is an optional ledger linked to a primary ledger for the purpose of tracking alternative accounting. A secondary ledger can differ from its primary ledger by using a different accounting method, chart of accounts, accounting calendar, currency, or processing options. All or some of the journal entries processed in the primary ledger are transferred to the secondary ledger, based on your configuration options. The transfers are completed based on the conversion level selected. There are four conversion levels:

- **Balance**: Only Oracle Fusion General Ledger balances are transferred to the secondary ledger.
- **Journal**: General Ledger journal posting process transfers the journal entries to the secondary ledger.
- **Subledger**: Oracle Fusion Subledger Accounting creates subledger journals to subledger level secondary ledgers as well as reporting currencies.
• Adjustments Only: Incomplete accounting representation that only holds adjustments. The adjustments can be manual or detailed adjustments from Subledger Accounting. This type of ledger must share the same chart of accounts, accounting calendar, and period type combination, and currency as the associated primary ledger.

**Note**

A full accounting representation of your primary ledger is maintained in any subledger level secondary ledger.

Secondary ledgers provide functional benefits, but produce large volumes of additional journal entry and balance data, resulting in additional performance and memory costs. When adding a secondary ledger, consider your needs for secondary ledgers or reporting currencies, and select the least costly data conversion level that meets your requirements. For secondary ledgers, the least costly level is the adjustment data conversion level because it produces the smallest amount of additional data. The balance data conversion level is also relatively inexpensive, depending upon how often the balances are transferred from the primary to the secondary ledger. The journal and subledger data conversion levels are much more expensive, requiring duplication of most general ledger and subledger journal entries, as well as general ledger balances.

For example, you maintain a secondary ledger for your International Financial Reporting Standards (IFRS) accounting requirements, while your primary ledger uses US Generally Accepted Accounting Principles (GAAP). You decided to select the subledger level for your IFRS secondary ledger. However, since most of the accounting is identical between US GAAP and IFRS, a better solution is to use the adjustment only level for your secondary ledger. The subledger level secondary ledger requires duplication of most subledger journal entries, general ledger journal entries, and general ledger balances. With the adjustment only level, your secondary ledger contains only the adjustment journal entries and balances necessary to convert your US GAAP accounting to the IFRS accounting, which uses a fraction of the resources that are required by full subledger level secondary ledger.

Following are scenarios that may require different combinations of primary and secondary ledgers:

• The primary and secondary ledgers use different charts of accounts to meet varying accounting standards or methods. A chart of accounts mapping is required to instruct the application how to propagate balances from the source (primary) chart of accounts to the target (secondary) chart of accounts.

• The primary and secondary ledgers use different accounting calendars to comply with separate industry and corporate standards.

**Note**

Use the same currency for primary and secondary ledgers to avoid difficult reconciliations, if you have the resources to support the extra posting time and data storage. Use reporting currencies or translations to generate the different currency views needed to comply with internal reporting needs and consolidations.
Reporting Currencies

Reporting currencies maintain and report accounting transactions in additional currencies. Each primary and secondary ledger is defined with a ledger currency that is used to record your business transactions and accounting data for that ledger. It is advisable to maintain the ledger in the currency in which the majority of its transactions are denominated. For example, create, record, and close a transaction in the same currency to save processing and reconciliation time. Compliance, such as paying local transaction taxes, is also easier using a local currency. Many countries require that your accounting records be kept in their national currency.

If you need to maintain and report accounting records in different currencies, you do this by defining one or more reporting currencies for the ledger. There are three conversion levels for reporting currencies:

- **Balance**: Only General Ledger balances are converted into the reporting currency using translation.
- **Journal**: General Ledger journal entries are converted to the reporting currency during posting.
- **Subledger**: Subledger Accounting creates subledger reporting currency journals along with primary ledger journals.

**Note**

A full accounting representation of your primary ledger is maintained in any subledger level reporting currency. Secondary ledgers cannot use subledger level reporting currencies.

Of the three data conversion levels available, the balance data conversion level is typically the least expensive, requiring duplication of only the balance level information. The journal and subledger data conversion levels are more expensive, requiring duplication of most general ledger and subledger journal entries, as well as general ledger balances.

Do not use journal or subledger level reporting currencies if your organization has only an infrequent need to translate your financial statements to your parent company’s currency for consolidation purposes. Standard translation functionality meets this need. Consider using journal or subledger level reporting currencies when any of the following conditions exist.

- You operate in a country whose unstable currency makes it unsuitable for managing your business. As a consequence, you need to manage your business in a more stable currency while retaining the ability to report in the unstable local currency.
- You operate in a country that is part of the European Economic and Monetary Union (EMU), and you choose to account and report in both the European Union currency and your National Currency Unit (NCU).

**Note**

The second option is rare since most companies have moved beyond the initial conversion to the EMU currency. However, future decisions could add other countries to the EMU, and then, this option would again be used during the conversion stage.
Financial Ledgers: How They Fit Together

Oracle Fusion Applications is an integrated suite of business applications that connects and automates the entire flow of the business process across both front and back office operations and addresses the needs of a global enterprise. The process of designing the enterprise structure, including the accounting configuration, is the starting point for an implementation. This process often includes determining financial, legal, and management reporting requirements, setting up primary and secondary ledgers, making currency choices, and examining consolidation considerations.

This figure shows the enterprise structure components and their relationships to each other. Primary ledgers are connected to reporting currencies and secondary ledgers to provide complete reporting options. Legal entities are assigned to ledgers, both primary and secondary, and balancing segments are assigned to legal entities. Business units must be connected to both a primary ledger and a default legal entity. Business units can record transactions across legal entities.

Primary Ledgers

A primary ledger is the main record-keeping ledger. Create a primary ledger by combining a chart of accounts, accounting calendar, ledger currency, and accounting method. To determine the number of primary ledgers, your enterprise structure analysis must begin with determining financial, legal, and management reporting requirements. For example, if your company has separate subsidiaries in several countries worldwide, create multiple primary ledgers.
representing each country with the local currency, chart of accounts, calendar, and accounting method to enable reporting to each country’s legal authorities. If your company just has sales in different countries, with all results being managed by the corporate headquarters, create one primary ledger with multiple balancing segment values to represent each legal entity. Use secondary ledgers or reporting currencies to meet your local reporting requirements, as needed. Limiting the number of primary ledgers simplifies reporting because consolidation is not required. Other consideration such as corporate year end, ownership considerations, and local government regulations, also affect the number of primary ledgers required.

Secondary Ledgers
A secondary ledger is an optional ledger linked to a primary ledger. A secondary ledger can differ from its related primary ledger in chart of accounts, accounting calendar, currency, accounting method, or ledger processing options. Reporting requirements, for example, that require a different accounting representation to comply with international or country-specific regulations, create the need for a secondary ledger.

Below are scenarios and required action for different components in primary and secondary ledgers:

- If the primary and secondary ledgers use different charts of accounts, the chart of accounts mapping is required to instruct the system how to propagate journals from the source chart of accounts to the target chart of accounts.
- If the primary and secondary ledgers use different accounting calendars, the accounting date and the general ledger date mapping table will be used to determine the corresponding non-adjusting period in the secondary ledger. The date mapping table also provides the correlation between dates and non-adjusting periods for each accounting calendar.
- If the primary ledger and secondary ledger use different ledger currencies, currency conversion rules are required to instruct the system on how to convert the transactions, journals, or balances from the source representation to the secondary ledger.

Note: Journal conversion rules, based on the journal source and category, are required to provide instructions on how to propagate journals and types of journals from the source ledger to the secondary ledger.

Reporting Currencies
Reporting currencies are the currency you use for financial, legal, and management reporting. If your reporting currency is not the same as your ledger currency, you can use the foreign currency translation process or reporting currencies functionality to convert your ledger account balances in your reporting currency. Currency conversion rules are required to instruct the system on how to convert the transactions, journals, or balances from the source representation to the reporting currency.

Legal Entities
Legal entities are discrete business units characterized by the legal environment in which they operate. The legal environment dictates how the legal entity should perform its financial, legal, and management reporting. Legal entities generally have the right to own property and the obligation to comply with
Balancing Segments

You assign primary balancing segment values to all legal entities before assigning values to the ledger. Then, assign specific primary balancing segment values to the primary and secondary ledgers to represent nonlegal entity related transactions such as adjustments. You can assign any primary balancing segment value that has not already been assigned to a legal entity. You are allowed to assign the same primary balancing segment values to more than one ledger. The assignment of primary balancing segment values to legal entities and ledgers is performed within the context of a single accounting setup. The Balancing Segment Value Assignments report is available to show all primary balancing segment values assigned to legal entities and ledgers across accounting setups to ensure the completeness and accuracy of their assignments. This report allows you to quickly identify these errors and view any unassigned values.

Business Units

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. When a business function produces financial transactions, a business unit must be assigned 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. Normally, it will have a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss. You define business units as separate task generally done after the accounting setups steps.

The business unit model:

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

For example, if your company requires business unit managers to be responsible for managing all aspects of their part of the business, then consider using two balancing segments, company and business unit to enable the production of business unit level balance sheets and income statements.

Transactions are exclusive to business units. In other words, you can use business unit as a securing mechanism for transactions. For example, if you have an export business that you run differently from your domestic business, use business units to secure members of the export business from seeing the transactions of the domestic business.

Creating Primary Ledgers: Example

Create a primary ledger as your main record-keeping ledger. Like any other ledger, a primary ledger records transactional balances by using a chart of accounts with a calendar, currency, and accounting rules implemented in an
accounting method. The primary ledger is closely associated with the subledger transactions and provides context and accounting for them.

**Scenario**

Your company, InFusion Corporation is implementing Oracle Fusion Applications. You have been assigned the task of creating a primary ledger for your InFusion America entity.

1. Navigate to the Define Accounting Configurations task list and open Manage Primary Ledgers from within your implementation project. Click the Go to Task.
2. Click the Create icon.
3. Enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>InFusion America</td>
</tr>
<tr>
<td>Description</td>
<td>InFusion America primary ledger for recording transactions.</td>
</tr>
<tr>
<td>Chart of Accounts</td>
<td>InFusion America Chart of Accounts</td>
</tr>
<tr>
<td>Accounting Calendar</td>
<td>Standard Monthly</td>
</tr>
<tr>
<td>Currency</td>
<td>USD</td>
</tr>
<tr>
<td>Accounting Method</td>
<td>Standard Accrual</td>
</tr>
</tbody>
</table>

4. Click Save and Edit Task List to navigate back to the accounting configuration task list.

**Note**

You cannot change the chart of accounts, accounting calendar, or currency for your ledger after you save your ledger.

**Define Accounting Configurations of Enterprise Structures: Specify Ledger Options**

**Specifying Ledger Options: Worked Example**

This example demonstrates specifying the ledger options for your primary ledger. Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion enterprise resource planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers.

After completing your InFusion America Primary Ledger, select Specify Ledger Options under the Define Accounting Configuration task list on the Functional Setup Manager page.
Note

Both primary and secondary ledgers are created in the same way and use the same user interface to enable their specific ledger options.

Reviewing General Region Options
1. Accept the Name and Description defaults for the ledger selected.
2. Review the Currency and Chart of Accounts for the specified ledger, which are automatically populated.

Setting Accounting Calendar Region Options
1. Review the Accounting Calendar that defaults from your ledger.
2. Select Jan-2011 as the First Open Period for your ledger.
   Important: Select a period after the first defined period in the ledger calendar to enable running translation. You cannot run translation in the first defined period of a ledger calendar. In this example, your calendar began with Jan-2010.
3. Enter 3 for the Number of Future Enterable Periods.
   Any value between 0 and 999 periods can be specified to permit entering journals but not posting them in future periods. Minimize the number of open and future periods to prevent entry in the wrong period.

Selecting the Subledger Accounting Region Options
1. Accept the default Accounting Method from your ledger.

Completing the Period Close Region Options
1. Enter your Retained Earnings Account:
   101-00-31330000-0000-0000-0000-0000.
   This account is required for the General Ledger to perform the movement of revenue and expense account balances to this account at the end of the accounting year.
2. Enter your Cumulative Translation Adjustment Account:
   101-00-31350000-0000-0000-0000-0000.
   Note: The Cumulative Translation Adjustment (CTA) account is required for ledgers running translation.
3. Do not enter a Default Period End Rate Type or Default Period Average Rate Type.
   The values entered here are used as the default for balance level reporting currency processing. InFusion America Primary Ledger is using the subledger level reporting currency processing.

Specifying the Journal Processing Region Options
1. Specify the Balance options as outlined in the following table.
### Option Setting Table

<table>
<thead>
<tr>
<th>Option</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Suspense</td>
<td>General Ledger</td>
</tr>
<tr>
<td>Default Expense Account</td>
<td>101-00-98199999-0000-0000-0000-0000</td>
</tr>
<tr>
<td>Rounding Account</td>
<td>101-10-98189999-0000-0000-0000-0000</td>
</tr>
<tr>
<td>Entered Currency Balancing Account</td>
<td>101-10-98179999-0000-0000-0000-0000</td>
</tr>
<tr>
<td>Balancing Threshold Percent</td>
<td>10</td>
</tr>
</tbody>
</table>

2. Click all the following Entry options listed in the table.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable journal approval</td>
<td>Click to enable journal approval functionality. Approval rules must be created in the Oracle Fusion Approvals Management (AMX).</td>
</tr>
<tr>
<td>Notify when prior period journal</td>
<td>Notify the user when a prior period date is selected on a journal entry.</td>
</tr>
<tr>
<td>Allow mixed and statistical journals</td>
<td>Enter both monetary and statistical amounts on the same line in a journal entry.</td>
</tr>
<tr>
<td>Validate reference date</td>
<td>Requires a reference date in an open or future enterable period.</td>
</tr>
</tbody>
</table>

3. Click the **Separate journals by accounting date during journal import** for the Import option to create individual journal entries for each accounting date.

4. For the Reversal options, select InFusion America Accrual Set from the list of values in the **Journal Reversal Criteria Set** field and click the **Launch AutoReverse after open period** to reverse accrual journal entries automatically when a new period is opened.

5. Click the **Enable intercompany accounting** for the Intercompany option to enable automatic balancing by the application for primary, second, and third balancing segments (if implemented) on intercompany journal entries and transactions.

Note: To complete the intercompany accounting functionality, you must define intercompany rules.

### FAQs for Specify Ledger Options

#### What happens if I change the cumulative adjustment account?

To avoid data corruption, your cumulative adjustment account (CTA) can only be changed if you first perform the following set of steps:

- Purge all translated balances
• Change the CTA account
• Rerun translation

What happens if I change the retained earnings account?

To avoid data corruption, your retained earnings account can only be changed if you first perform the following set of steps:

• Enter and post journals to bring the ending balances for your income statement accounts to zero at the end of each accounting year
• Purge actual translated balances
• Update the retained earnings account
• Reverse the journal entries used to bring the ending account balances to zero and rerun translation

Assigning Legal Entities and Balancing Segments: Examples

Optionally, assign legal entities and balancing segments to your accounting configuration.

Assign Legal Entities
Assign one or more legal entities to your configuration by following these steps:

1. Navigate to the Assign Legal Entities task. Click the Go to Task.
2. Click the Select and Add icon.
3. Click Search. Select your legal entities.
4. Click Apply. Click Done.
5. Click Save and Close.

Assign Balancing Segments to Legal Entities
Assign balancing segment values to your legal entities by following these steps:

1. Navigate to the Assign Balancing Segment Values to Legal Entities task. Click the Go to Task.
2. Click the Create icon.
3. Select the balancing segment value. Optionally, add a Start Date.
4. Click Save and Close to close the create page.
5. Click Save and Close.

Assign Balancing Segments to Ledgers
Assign balancing segment values directly to your ledger by following these steps:

1. Navigate to the Assign Balancing Segment Value to Ledger task. Click the Go to Task.
2. Select the balancing segment value.
3. Optionally enter a start date.

4. Click **Save and Close**.

**Note**

The balancing segment values that are assigned to the ledger represent nonlegal entity transactions, such as adjustments. If you use legal entities, you must assign balancing segment values to all legal entities before assigning values to the ledger. The only available balancing segment values that can be assigned to ledgers are those not assigned to legal entities.

---

**Define Accounting Configurations of Enterprise Structures: Manage Reporting Currencies**

**Reporting Currency Balances: How They Are Calculated**

Reporting currency balances, set at the journal or subledger level, are updated when journal entries that originate in Oracle Fusion General Ledger are posted and converted to your reporting currencies. This process includes General Ledger manual journals, periodic journals, and allocations, and at the subledger level, journals from Oracle Fusion Subledger Accounting and imported from sources other than your Oracle Fusion subledgers. When you post a journal in a ledger that has one or more reporting currencies defined, the posting process creates new journals converted to each of your reporting currencies and includes them in the same batch as the original journal with a status of Posted.

**Settings That Affect**

Reporting currencies share a majority of the ledger options with their source ledger. For example, the reporting currency uses the same suspense account and retained earnings accounts as its source ledger. However, there are certain options that need to be set specifically for the reporting currencies. For example, reporting currencies are maintained at one of these three currency conversion levels:

- **Balance Level**: Only balances are maintained in the reporting currency using the General Ledger Translation process.
- **Journal Level**: Journal entries and balances are converted to the reporting currency by the General Ledger Posting process.
- **Subledger Level**: Subledger Accounting creates reporting currency journals for subledger transactions. General Ledger converts journals that originated in General Ledger or that are imported from sources other than the Oracle Fusion subledgers. The full accounting representation of your primary ledger is maintained in the subledger level reporting currency.

**Note**

Secondary Ledgers cannot use subledger level reporting currencies.

There are multiple dependencies between a reporting currency and its source ledger. Therefore, it is important that you complete your period opening tasks.
daily journal or subledger level reporting currencies accounting tasks, and period closing tasks in the correct order. Some guidelines are presented in the table below.

<table>
<thead>
<tr>
<th>Type</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period Opening Tasks</td>
<td>Open the accounting period in both your ledger and reporting currencies before you create or import journals for the period. Converted journals are only generated in your reporting currency if the period is open or future enterable.</td>
</tr>
<tr>
<td>Daily Tasks</td>
<td>Enter the daily conversion rates to convert your journals to each of your reporting currencies.</td>
</tr>
</tbody>
</table>
| Period Closing Tasks        | • Finish entering all regular and adjusting journals for the period in your ledger.  
• Post all unposted journals in your ledger if not already done in the previous step.  
• Post all unposted journals in your reporting currencies if not already done in the previous step.  
• Run Revaluation in both your ledger and reporting currencies. Post the resulting revaluation batches in each ledger.  
• As needed, translate balances in your ledger.  
• Generate needed reports from both your ledger and reporting currencies.  
• Close your accounting period in both your ledger and reporting currencies. |

How Reporting Currencies Are Calculated

If you use reporting currencies at the journal or subledger level, when you create accounting, post journal entries, or translate balances, journals are posted in your reporting currency. General Ledger and Subledger Accounting automatically generate journals in your reporting currencies where the entered currency amounts are converted to the reporting currency amounts. Other factors used in the calculation of reporting currency balances are listed:

• Manual Journals: Enter a manual journal batch in your reporting currency at the journal or subledger level by using the Create Journals page. Select the journal or subledger level reporting currency from the ledger’s list of values and continue in the same manner as entering any other manual journal.

• Conversion Rounding: Use the reporting currency functionality to round converted and accounted amounts using the same rounding rules used throughout your Oracle Fusion Applications. The reporting currency functionality considers several factors that are a part of the currencies predefined in your applications, including:

• Currency Precision: Number of digits to the right of the decimal point used in currency transactions.
• Minimum Accountable Unit: Smallest denomination used in the currency. This might not correspond to the precision.

• Converted Journals: Generate and post automatically, using the General Ledger Posting process, journals in your reporting currencies when you post the original journals in the source ledger for the following types of journals:
  • Manual journals
  • Periodic and allocation journals
  • Unposted journals from non-Oracle subledger applications
  • Unposted journals from any Oracle Fusion subledger that does not support reporting currency transfer and import
  • Optionally, revaluation journals

• Unconverted Journals: Rely on the subledger accounting functionality to converted and transfer Oracle Fusion subledger journals for both the original journal and the reporting currency journal to the General Ledger for import and posting. The reporting currency conversion for these journals is not performed by the General Ledger.

• Approving Journals: Use the journal approval feature to process reporting currency journals through your organization’s approval hierarchy. You can enable journal approval functionality separately in your source ledger and reporting currencies.

• Document Numbers: Accept the default document numbers assigned by the General Ledger application to your journal when you enter a journal in your ledger. The converted journal in the reporting currency is assigned the same document number. However, if you enter a journal in the reporting currency, the document number assigned to the journal is determined by the reporting currency.

• Sequential Numbering: Enable sequential numbering if you want to maintain the same numbering in your reporting currency and source ledger for journals, other than those journals for Oracle Fusion subledgers. Do not create separate sequences for your reporting currencies. If you do, the sequence defined for the reporting currencies is used and can cause document numbers not to be synchronized between the ledger and reporting currencies.

Note
If the Sequential Numbering profile option is set to **Always Used** or **Partially Used** and you define an automatic document numbering sequence, General Ledger enters a document number automatically when you save your journal. If you use manual numbering, you can enter a unique document number.

• Revaluation: Run periodically revaluation in your ledger and reporting currencies as necessary to satisfy the accounting regulations of the country in which your organization operates.

• Account Inquiries: Perform inquires in the reporting currency. Drill down to the journal detail that comprises the reporting currency balance. If the
journal detail is a converted journal that was converted automatically when the original journal was posted in the source ledger, you can drill down further to see the source ledger currency journal amounts.

**Note**

Be careful when changing amounts in a reporting currency, since the changes are not reflected in your source ledger. Making journal entry changes to a reporting currency makes it more difficult to reconcile your reporting currency to your source ledger. In general, enter or change your journals in your source ledger, and then allow posting to update the reporting currency.

**Note**

If you use reporting currencies at the journal or subledger level, statistical journals are generated for your reporting currencies, but the journals are not affected by the currency conversion process.

### Define Business Units: Manage Business Units

#### 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.

**Define Business Units: Assign Business Unit Business Function**

**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 Business Units: Manage Service Provider Relationships

Shared Service Centers: Explained

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

Service Provider Model

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

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

**Shared Service Center: Points to Consider**

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

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

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

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

**Security**

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

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

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

**Benefits**

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

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

**Service Provider Model: Explained**

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

**Procurement Example**

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

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

Customer Contracts Business Unit Setup: Explained

Using the Specify Customer Contract Management Business Function Properties task, available by navigating to Setup and Maintenance work area and searching on the task name, you can specify a wide variety of business function settings for customer contracts in a specific business unit. The selections you make for these business functions impact how Oracle Fusion Enterprise Contracts behaves during contract authoring.

Using the Specify Customer Contract Management Business Function Properties task, manage these business function properties:

- Enable related accounts
- Set currency conversion details
- Manage project billing options
- Set up clause numbering
- Set up the Contract Terms Library

The setup options available for the Contract Terms Library are applicable to both customer and supplier contracts, and are described in the business unit setup topic for the Contract Terms Library. That topic is available as a related link to this topic.

Enabling Related Customer Accounts

Contract authors can specify bill-to, ship-to, and other accounts for the parties in a contract. Enable the related customer accounts option if you want accounts previously specified as related to the contract party to be available for selection.
Managing Currency Conversion Options

If your organization plans to transact project-related business in multiple currencies, then select the multicurrency option. This allows a contract author to override a contract’s currency, which defaults from the ledger currency of the business unit. It also enables the contract author to specify currency conversion attributes to use when converting from the bill transaction currency to the contract currency and from the invoice currency to the ledger currency.

In the Bill Transaction Currency to Contract Currency region, enter currency conversion details that will normally be used, by all contracts owned by this business unit, to convert transaction amounts in the bill transaction currency to the contract currency. Newly created contracts contain the default currency conversion values, but you can override the values on any contract, if needed.

In the Invoice Currency to Ledger Currency region:

- Enter invoice transaction conversion details if the invoice and ledger currencies can be different.
- Enter revenue transaction conversion details if the revenue and ledger currencies can be different for as-incurred and rate-based revenue.

Managing Project Billing Options

The options available for selection in the Project Billing region control the behavior of project invoicing and revenue recognition for contracts with project-based work.

Project billing can behave differently for external contracts (customer billing) or intercompany and interproject contracts (internal billing).

Set these options, which apply to all contracts:

- Select the Transfer Revenue to General Ledger option if you want to create revenue accounting events and entries, and transfer revenue journals to the general ledger. If this option is not selected, then revenue can still be generated, but will not be transferred to the general ledger.
- Indicate if a reason is required for credit memos that are applied to invoices.

There are two sets of the following options, one for customer billing and a second for internal billing:

- Select an invoice numbering method, either Manual or Automatic. The invoice numbering method is the method that Oracle Fusion Receivables uses to number its invoices, upon release of draft invoices from Project Billing.
  - If the invoice numbering method is Manual, then select an invoice number type, which sets the type of Receivables invoice numbers that are allowed. Valid values are Alphanumeric and Numeric.
  - If the invoice numbering method is Automatic, then enter the next invoice number to use when generating Receivables invoice numbers.
- Select the Receivables batch source to use when transferring invoices to Receivables.

Set this option only for customer billing:
• Indicate if you want contract authors to manually enter the Receivables transaction type on the customer contracts they create.

Managing Clause Numbering
You can choose to number clauses manually or automatically.
If you choose the automatic numbering method, you must select a determinant level for the numbering. You must then select the appropriate clause sequence category from document sequences that you set up for this numbering level.

Contract Terms Library Business Unit Setup: Explained

You can specify a wide variety of Contract Terms Library settings for either customer or supplier contracts within each business unit, by using either the Specify Customer Contract Management Business Function Properties or the Specify Supplier Contract Management Business Function Properties tasks. These tasks are available by navigating to the Setup and Maintenance work area and searching on the task name.

For the Contract Terms Library in each business unit, you can:
• Enable clause and template adoption.
• Set the clause numbering method.
• Set the clause numbering level for automatic clause numbering of contracts.
• For a contract with no assigned ledger or legal entity, set the document sequence to Global or Business Unit level.
• Enable the Contract Expert feature.
• Specify the layout for printed clauses and contract deviation reports.

Enabling Clause Adoption
If you plan to use clause adoption in your implementation, then set up the following:
• Specify a global business unit
  You must designate one of the business units in your organization as the global business unit by selecting the Global Business Unit option. This makes it possible for the other local business units to adopt and use approved content from that global business unit. If the Global Business Unit option is not available for the business unit you are setting up, this means that you already designated another business unit as global.
• Enable automatic adoption
  If you are implementing the adoption feature, then you can have all the global clauses in the global business unit automatically approved and available for use in the local business by selecting the Autoadopt Global Clauses option. If you do not select this option, the employee designated as the Contract Terms Library Administrator must approve all global clauses before they can be adopted and used in the local business unit. This option is available only for local business units.
• Specify the administrator who approves clauses available for adoption
You must designate an employee as the Contract Terms Library administrator if you are using adoption. If you do not enable automatic adoption, then the administrator must adopt individual clauses or localize them for use in the local business unit. The administrator can also copy over any contract terms templates created in the global business unit. The clauses and contract terms templates available for adoption are listed in the administrator’s Terms Library work area.

**Setting Clause Numbering Options**

You can set up automatic clause numbering for the clauses in the business unit by selecting Automatic in the **Clause Numbering** field and setting the clause numbering level. Then select the appropriate clause sequence category for the specified numbering level. You must have previously set up document sequences for the document sequence categories of global, ledger, and business unit. If clause numbering is manual, contract terms library administrators must enter unique clause numbers each time they create a clause.

You can choose to display the clause number in front of the clause title in contracts by selecting the **Display Clause Number in Clause Title** option.

**Enabling Contract Expert**

You must select the **Enable Contract Expert** option to be able to use the Contract Expert feature in a business unit. This setting takes precedence over enabling Contract Expert for individual contract terms templates.

**Specifying the Printed Clause and Deviations Report Layouts**

For each business unit, you can specify the Oracle BI Publisher RTF file that serves as the layout for:

- The printed contract terms
  
  Enter the RTF file you want used for formatting the printed clauses in the **Clause Layout Template** field.

- The contract deviations report
  
  The RTF file you select as the **Deviations Layout Template** determines the appearance of the contract deviations report PDF. This PDF is attached to the approval notification sent to contract approvers.

**Define Business Units: Specify Supplier Contract Management Business Function Properties**

**Supplier Contracts Business Unit Setup: Explained**

Using the **Specify Supplier Contract Management Business Function Properties** task, available by selecting Setup and Maintenance from the Tools menu and searching on the task name, you can specify a variety of business function settings for supplier contracts in a specific business unit.
The selections you make for these business functions impact how the Contract Terms Library behaves during supplier contract authoring.

**Managing Contract Terms Library Setup Options**

The setup options available for the Contract Terms Library are applicable to both customer and supplier contracts, and are described in the business unit setup topic for the Contract Terms Library. That topic is available as a related link to this topic.

**Contract Terms Library Business Unit Setup: Explained**

You can specify a wide variety of Contract Terms Library settings for either customer or supplier contracts within each business unit, by using either the Specify Customer Contract Management Business Function Properties or the Specify Supplier Contract Management Business Function Properties tasks. These tasks are available by navigating to the Setup and Maintenance work area and searching on the task name.

For the Contract Terms Library in each business unit, you can:

- Enable clause and template adoption.
- Set the clause numbering method.
- Set the clause numbering level for automatic clause numbering of contracts.
- For a contract with no assigned ledger or legal entity, set the document sequence to Global or Business Unit level.
- Enable the Contract Expert feature.
- Specify the layout for printed clauses and contract deviation reports.

**Enabling Clause Adoption**

If you plan to use clause adoption in your implementation, then set up the following:

- Specify a global business unit
  
  You must designate one of the business units in your organization as the global business unit by selecting the Global Business Unit option. This makes it possible for the other local business units to adopt and use approved content from that global business unit. If the Global Business Unit option is not available for the business unit you are setting up, this means that you already designated another business unit as global.

- Enable automatic adoption
  
  If you are implementing the adoption feature, then you can have all the global clauses in the global business unit automatically approved and available for use in the local business by selecting the Autoadopt Global Clauses option. If you do not select this option, the employee designated as the Contract Terms Library Administrator must approve all global clauses before they can be adopted and used in the local business unit. This option is available only for local business units.
• Specify the administrator who approves clauses available for adoption

You must designate an employee as the Contract Terms Library administrator if you are using adoption. If you do not enable automatic adoption, then the administrator must adopt individual clauses or localize them for use in the local business unit. The administrator can also copy over any contract terms templates created in the global business unit. The clauses and contract terms templates available for adoption are listed in the administrator’s Terms Library work area.

**Setting Clause Numbering Options**

You can set up automatic clause numbering for the clauses in the business unit by selecting Automatic in the **Clause Numbering** field and setting the clause numbering level. Then select the appropriate clause sequence category for the specified numbering level. You must have previously set up document sequences for the document sequence categories of global, ledger, and business unit. If clause numbering is manual, contract terms library administrators must enter unique clause numbers each time they create a clause.

You can choose to display the clause number in front of the clause title in contracts by selecting the **Display Clause Number in Clause Title** option.

**Enabling Contract Expert**

You must select the **Enable Contract Expert** option to be able to use the Contract Expert feature in a business unit. This setting takes precedence over enabling Contract Expert for individual contract terms templates.

**Specifying the Printed Clause and Deviations Report Layouts**

For each business unit, you can specify the Oracle BI Publisher RTF file that serves as the layout for:

- The printed contract terms
  Enter the RTF file you want used for formatting the printed clauses in the **Clause Layout Template** field.

- The contract deviations report
  The RTF file you select as the **Deviations Layout Template** determines the appearance of the contract deviations report PDF. This PDF is attached to the approval notification sent to contract approvers.

**Define Workforce Structures: Manage Locations**

**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.

**Define Workforce Structures: FAQs for Manage Locations**

**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.

Define Workforce Structures: Manage Divisions

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>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>
<tr>
<td>How many legal entities?</td>
<td>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.</td>
</tr>
<tr>
<td>Create legislative data group?</td>
<td>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.</td>
</tr>
</tbody>
</table>

**Note**
When you identify the legal entity as a payroll statutory unit, the application transfers the legal reporting unit that is associated with that legal entity to Oracle Fusion HCM as a tax reporting unit.
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.

Define Workforce Structures: Manage Departments

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 Sales Cloud, 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 Oracle Sales Cloud 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.

Define Workforce Structures: FAQs for Manage Job Families

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 Workforce Structures: Manage Job

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.

![Software Industry Job Setup Diagram](image)

**Define Workforce Structures: Manage Person Search Relevance Profile Option Values**

**Search Relevance Profile Options: Explained**

The strength of the relationship between the person performing a gallery search and each person whose assignment appears in the search results can determine the order of the results: the stronger the relationship, the closer to the top of the
results an assignment appears. The search relevance profile options control how the strength of the relationship between the searcher and the search result is calculated.

**Weighting Profile Options**

Using the following profile options, you can change the weighting applied to the relevant factors.

<table>
<thead>
<tr>
<th>Profile Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HR: Organization Hierarchy Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the organization hierarchy proximity factor.</td>
</tr>
<tr>
<td>HR: Position Hierarchy Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the position hierarchy proximity factor.</td>
</tr>
<tr>
<td>HR: Manager Hierarchy Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the manager hierarchy proximity factor.</td>
</tr>
<tr>
<td>HR: Location Proximity Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the location proximity factor.</td>
</tr>
<tr>
<td>HR: Selection History Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the selection history factor.</td>
</tr>
<tr>
<td>HR: Social Network Weight</td>
<td>Specifies the weighting applied to the relationship strength value for the social network factor.</td>
</tr>
</tbody>
</table>

The default value of each weighting profile option is 0.5. To increase the relevance of a factor relative to other factors, you increase its weighting; to decrease its relevance, you reduce its weighting.

**HR: Selection History Timeout**

The number of times the searcher selects a person's assignment from the search results during a specified period, which is 7 days by default, is recorded automatically. You can specify this period for the enterprise on the HR: Selection History Timeout profile option.

**HR: Maximum Hierarchy Proximity**

When the searcher's primary assignment is in the same organization, position, or manager hierarchy as a person's assignment, the strength of the relationship depends on their proximity to each other in the hierarchy. The maximum number of hierarchy boundaries to include in the calculation is 4 by default. You can set this value for the enterprise on the HR: Maximum Hierarchy Proximity profile option.

**HR: Relationship Priority Factor**

The searcher can specify a rating for a search result, and each rating is associated with a multiplying factor. On this profile option, you can specify the highest possible multiplying factor that can be applied to a search result. By default, the multiplying factor is 2. If you increase its value, you increase the significance of the searcher's own ratings relative to other factors.
Define Facilities: Manage Facility Shifts, Workday Patterns, and Schedules

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.

Define Facilities: Manage Inventory Organizations

Inventory Organizations: Explained

An inventory organization is a logical or physical entity in the enterprise that is used to store definitions of items or store and transact items.

You select the following usages in the inventory organization's properties:

- Item management
- Item and inventory management

Item Management

Inventory organizations used for item management, which are the same as item organizations, store only definitions of items. Use inventory organizations for item management when the storage or movement of inventory does not need to be physically or financially tracked. For example, in a retail implementation you can create an inventory organization for item management to store the names of items that are listed by and sold through each retail outlet, while a different system tracks physical inventory and transactions. If it is necessary in the future, you can change an inventory organization's usage from item management to item and inventory management in the inventory organization's properties.

Item and Inventory Management

Inventory organizations used for item and inventory management store and transact items, in addition to item definitions. An inventory organization used for item and inventory management is associated with one business unit, one legal entity, and one primary ledger. Use inventory organizations for item and inventory management when the storage or movement of inventory needs to be physically and financially tracked. Inventory organizations used for item and inventory management can represent facilities such as manufacturing centers, warehouses, or distribution centers. You cannot change an inventory organization's use from item and inventory management to item management.

Inventory Organization: Critical Choices

In Oracle Fusion, storage facilities, warehouses, and distribution centers are implemented as inventory organizations.

Inventory organizations are:

- Managed by a business unit, with the materials management business function enabled.
• Mapped to a legal entity and a primary ledger.

There are two types of inventory organizations:

• Manufacturing facilities

• Storage facilities

Storage and manufacturing facilities are related to other organizational entities through a business unit that stores, manufactures, and distributes goods through many factories, warehouses, and distribution centers. The material parameters are set for both the facilities, enabling movement of material in the organization. This business unit has the business function of Materials Management enabled. Oracle Fusion Applications allow many inventory organizations to be assigned to one business unit.

**Note**

Currently, Oracle Fusion Applications do not include manufacturing capabilities, so setup your manufacturing facilities outside of Oracle Fusion applications.

**Distribution Center as an Inventory Organization**

A distribution center can store inventory that is the responsibility of different business units. In this situation, assign an inventory organization to each business unit as a representation of the inventory in the distribution center. The multiple inventory organizations representing the inventory are defined with the same location to show that they are a part of the same distribution center.

In the following figure the two business units, Air Compressors and Air Transmission, share one distribution center in Atlanta. The two inventory organizations, Air Compressors and Air Transmission represent the inventory for each business unit in the Atlanta distribution center and are both assigned the Atlanta location.
Legal Entities Own Inventory Organizations

A legal entity owns the inventory located in a storage or manufacturing facility. This ownership is assigned through the relationship of the inventory organization representing the inventory and the legal entity assigned to the inventory organization. The legal entity assigned to the inventory organization shares the same primary ledger as the inventory organization’s business unit.

The inventory is tracked in the inventory organization owned by the legal entity of which the business unit is part. All transactions are accounted for in the primary ledger of the legal entity that owns the inventory.

The figure below illustrates the inventory owned by InFusion Air Quality legal entity. The InFusion Air Quality legal entity is associated with the Air Compressors business unit, which is associated with the two Air Compressors inventory organizations. Therefore, InFusion Air Quality legal entity owns the entire inventory in both the Dallas and Atlanta locations.

Facility Schedules Are Associated with Inventory Organizations

A prerequisite to defining an inventory organization is to define a facility schedule. Oracle Fusion Applications allow you to associate an inventory organization with a schedule.

Facility schedules allow creating workday calendars for inventory organizations that are used in the Oracle Fusion Supply Chain Management product family. For example, use workday calendars in the scheduling of cycle counts and calculating transit time.

Inventory Organization Prerequisites: Points to Consider

You can create a new inventory organization, or select an existing organization to define as an inventory organization.

Before creating inventory organizations:

- Set up inventory organization dependencies
- Plan inventory organization parameters
Setting Up Inventory Organization Dependencies

When you create an inventory organization, you must associate it to dependencies, such as business units and legal entities. For this reason, create these dependencies before creating an inventory organization.

Planning Inventory Organization Parameters

Before creating an inventory organization, plan the inventory organization’s parameters.

Consider the following when planning to configure an inventory organization’s parameters:

• Which schedule to use
• Which inventory organization to serve as the item master organization
• Whether to configure locator control and if so, the level at which to enforce the locator control
• How you want to configure movement request settings such as pick slip batch size and replenishment movement request grouping
  
Consider the size of your operation, your usage of subinventories, and the type of labor or equipment required when considering whether you want to use organization- or subinventory-level replenishment movement request grouping.

• How you want to configure lot, serial, and packing unit generation settings
  
To make appropriate choices for these settings, you should be familiar with:

• Your company’s guidelines for creating lot names, serial numbers, and packing unit numbers
• Whether your company requires you to assign the same lot number to multiple items in the same organization, or a specific lot number to only one item in the same organization
• Whether your company requires you to place purchase order or shipping order material under lot control
• How you want to configure item sourcing details, such as the picking rule to use, and whether to specify the inventory organization as a logistics services organization

Rounding the Reorder Quantity: How It Affects Min-Max Planning Reorder Calculations

When you specify to round reorder quantities, min-max planning reorders for item subinventories are automatically rounded up or down.
Settings That Affect Rounding the Reorder Quantity

Reorder quantities for an item subinventory are calculated based on:

- The setting that you select for the Round Order Quantity parameter on the Manage Inventory Organization Parameters page, General tab, of the inventory organization containing the item subinventory
- The value that you specify for the Fixed Lot Multiple text box on the Add Item to Subinventory window

How Rounding the Reorder Quantity Affects Min-Max Planning Reorder Quantity Calculations

If you enable rounding the reorder quantity for the inventory organization, and specify the fixed lot multiple for the item subinventory, the reorder quantity is rounded up. If you disable rounding the reorder quantity for the inventory organization, and specify the fixed lot multiple for the item subinventory, the reorder quantity is rounded down.

Note

To round reorder quantities, you must specify a fixed lot multiple.

Example: Rounding the Reorder Quantity

Assume that the reorder quantity is 24. If you enable rounding the reorder quantity and specify 10 for the fixed lot multiple, the reorder quantity is rounded up to 30. If you disable rounding the reorder quantity and keep the fixed lot multiple at 10, the reorder quantity is rounded down to 20.

Selecting Lot Number Uniqueness Control: Critical Choices

Select one of the following lot number uniqueness control options to apply to the items in your inventory organization:

- No uniqueness control
- Across items

No Uniqueness Control

You can assign the same lot number to multiple items in the same inventory organization and across inventory organizations. The following table provides an example of how lot numbers are generated when uniqueness control is not applied, both within and across inventory organizations.

<table>
<thead>
<tr>
<th>Within Inventory Organization</th>
<th>Across Inventory Organizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item AS100 (printer) / Lot LN100</td>
<td>Item AS100 (printer) / Lot LN100</td>
</tr>
<tr>
<td>Item AS101 (laptop computer) / Lot LN100</td>
<td>Item AS101 (laptop computer) / Lot LN100</td>
</tr>
</tbody>
</table>
Across Items

You can only assign a unique lot number to a single item in one inventory organization. If the same item is also in a different inventory organization, you must assign that item a unique lot number. The following table provides an example of how lot numbers are generated when uniqueness control is applied across items, both within and across inventory organizations.

<table>
<thead>
<tr>
<th>Within Inventory Organization</th>
<th>Across Inventory Organizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item AS100 (printer) / Lot LN100</td>
<td>Item AS100 (printer) / Lot LN300</td>
</tr>
<tr>
<td>Item AS101 (laptop computer) / Lot LN200</td>
<td>Item AS101 (laptop computer) / Lot LN400</td>
</tr>
</tbody>
</table>

FAQs for Manage Inventory Organizations

What happens if I select the Supplier item sourcing type for replenishment?

Items are replenished from an external supplier.

What happens if I create an inventory organization as a logistics services organization?

The inventory organization is not costed, and shipment lines from different logistics service provider customers cannot be packed in the same packing unit.

Define Facilities: Manage Item Organizations

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 Sales Cloud, 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.
Common Applications Configuration: Define Security for Procurement

Security Tasks: Highlights

Security tasks include the following.

- Security setup
- Security implementation and administration

Note

Security setup and administration tasks typically use integrated user interface pages that are provided by the following products.

- Oracle Identity Manager (OIM)
- Oracle Authorization Policy Manager (APM)
- Oracle Fusion Human Capital Management (HCM) products
- Oracle Application Access Control Governor (AACG) in Oracle Enterprise Governance, Risk and Compliance (GRC)

Security setup and administrative tasks performed by product administrators and implementation consultants, such as managing HCM security profiles, are presented in the documentation for those products.

Set Up the IT Security Manager Job Role

Provision the IT Security Manager job role with roles for user and role management.

- Using the OIM Administrator user name and password, sign in to Oracle Identity Manager (OIM). Refer to the Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management. See: Creating Users and Groups

- Open the IT Security Manager job role's attributes and use the Hierarchy tab to add the User Identity Administrators role and the Role Administrators role in the OIM Roles category using the Add action. Use the Delegated Administration menu to search for the Xellerate Users organization and assign it to the IT Security Manager role. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager. See: User Management Tasks

Prerequisite Tasks for Security Administration

Sign into Oracle Fusion Applications for the first time with the Installation Super User account to synchronize LDAP users with HCM user management and
create an IT security manager user account and provision it with the IT Security Manager role. For environments that are not in Oracle Cloud, use the super user account that was created during installation to sign in for the first time.

- Installation establishes the super user account. Refer to the Oracle Fusion Applications Installation Guide.

See: Identity Management Configuration

- Oracle provides an initial user for accessing your services in Oracle Cloud. For more information, refer to "Oracle Cloud Application Services Security: Explained" in Oracle Cloud documentation.

- Synchronize LDAP users with HCM user management by performing the Run User and Roles Synchronization Process task. Monitor completion of the predefined Enterprise Scheduler process called Retrieve Latest LDAP Changes.

- Refer to information about creating person records in Oracle Fusion Applications Workforce Development Implementation Guide, or refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

See: Managing Users

- As a security guideline, provision a dedicated security professional with the IT Security Manager role as soon as possible after initial security setup and revoke that role from users provisioned with the Application Implementation Consultant role. If entitled to do so, see Security Tasks and Oracle Fusion Applications: How They Fit Together for details about provisioning the IT security manager.

**Required Security Administration Tasks**

Establish at least one implementation user and provision that user with sufficient access to set up the enterprise for all integrated Oracle Fusion Middleware and all application pillars or partitions.

- Perform the initial security tasks. If entitled to do so, see Initial Security Administration: Critical Choices.

  - Sign in to Oracle Fusion Applications using the IT security manager’s or administrator’s user name and password, and create and provision users who manage your implementation projects and set up enterprise structures by performing the Create Implementation Users task. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

  See: User Management Tasks

  - Create a data role for implementation users who will set up HCM that grants access to data in secured objects required for performing HCM setup steps. Provision the implementation user with this View All data role. See "Creating an HCM Data Role: Worked Example."

  - For an overview of security tasks from the perspective of an applications administrator, refer to the Oracle Fusion Applications Administrator’s Guide

  See: Securing Oracle Fusion Applications
Optional Security Administration Tasks

Once initial security administration is complete and your enterprise is set up with structures such as business units, additional security administration tasks are optional and based on modifying and expanding the predefined security reference implementation to fit your enterprise. See points to consider for defining security, data security and trading partner security after enterprise setup.

- Create users. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.
  
  See: Creating Users

- Provision users with roles. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.
  
  See: Adding and Removing Roles

- You manage users and job roles, including data and abstract roles, in Oracle Identity Management user interface pages. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.
  
  See: User Interfaces

  
  See: Managing Oracle Fusion Applications Data Security Policies

- You manage role provisioning rules in Human Capital Management (HCM). Refer to the Role Mappings: Explained topic in the Oracle Fusion Applications Workforce Development Implementation Guide.
  

- For a complete description of the Oracle Fusion Applications security reference implementation, see the Oracle Fusion Applications Security Reference Manuals for each offering.
  

- For a detailed functional explanation of the Oracle Fusion Applications security approach, refer to the following guides.
  
  See: Oracle Fusion Applications Security Guide

See: Oracle Fusion Applications Security Hardening Guide

- Since security in Oracle Fusion Applications is based on integrations with Oracle Identity Management in Fusion Middleware, security features in the database, and Oracle Enterprise Governance, Risk and Compliance (GRC), additional resources in support of performing security tasks include the following.

  - Authorization Policy Manager (APM) is available in Oracle Fusion Applications through integration with Oracle Identity Management
Authorization policy management involves managing duty roles, data role templates, and data security policies. Refer to the Oracle Fusion Middleware Authorization Policy Manager Administrator's Guide.

See: Getting Started With Oracle Authorization Policy Manager

- Oracle Identity Management (OIM) is available in Oracle Fusion Applications through integration with Oracle Fusion Middleware. Identity management in Oracle Fusion Application involves creating and managing user identities, creating and linking user accounts, managing user access control through user role assignment, managing enterprise roles, and managing workflow approvals and delegated administration.

See: Oracle Fusion Middleware User's Guide for Oracle Identity Manager

- Oracle Fusion Applications is certified to integrate with Applications Access Controls Governor (AACG) in the Oracle Enterprise Governance, Risk and Compliance (GRC) suite to ensure effective segregation of duties (SOD).

See: Oracle Application Access Controls Governor Users Guide

See: Oracle Application Access Controls Governor Implementation Guide

- Configure and manage auditing. Refer to the Oracle Fusion Middleware Application Security Guide.

See: Configuring and Managing Auditing

Defining Security After Enterprise Setup: Points to Consider

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Import Worker Users
- Import Partner Users
- Manage Job Roles
- Manage Duties
- Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human Capital Management (HCM), Supplier Portal, and Oracle Sales Cloud Self Service, respectively, and associate them with users.
Before Importing Users

Oracle Identity Management (OIM) handles importing users. If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Initiate HCM Spreadsheet Load task. If user and role provisioning rules have been defined, the Initiate HCM Spreadsheet Load process automatically creates user and role provisioning requests as the workers are created. Once the enterprise is set up, performing the Initiate HCM Spreadsheet Load task populates the enterprise with HR workers in records linked by global user ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Initiate HCM Spreadsheet Load task results in new user accounts being created. Worker email addresses as an alternate input for the Initiate HCM Spreadsheet Load task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Initiate HCM Spreadsheet Load task in HCM.

Note

The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in Oracle Sales Cloud.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

Import Users

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks. If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Initiate HCM Spreadsheet Load task to create users from imported HR workers. Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager.
In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

Manage Job Roles

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

Note

Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, Oracle Sales Cloud, or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

Manage Duties

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering.

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.

In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.
Tip
As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

Note
Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

Manage Application Access Controls

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

Note
In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

Security Tasks and Oracle Fusion Applications: How They Fit Together

The major security tasks and their order within the context of an overall Oracle Fusion Applications implementation extend from security setup through production deployment audits.

The Oracle Fusion business process model (BPM) provides a sequence of security implementation tasks that includes the following.

- Security setup (Define Common Applications Configuration activity)
  - Define Implementation Users task group (optional)
    - Create Implementation Users task
    - Create Data Role for Implementation Users task
  - Provision Roles to Implementation Users task
• Define security - tasks vary depending on deployed Oracle Fusion product family
  • Revoke Data Role from Implementation Users task
  • Import Worker Users task
  • Import Partner Users task
  • Manage Duties task
  • Manage Job Roles task
  • Manage Application Access Controls task
  • Define Automated Governance, Risk, and Performance Controls activity
    • Manage Application Access Controls task (AACG settings)
    • Manage Application Preventive Controls task
    • Manage Application Transaction Controls task
    • Manage Application Configuration Controls task
• User and role provisioning tasks
  • Implement Role Request and Provisioning Controls activity
    • Import Worker Users task
    • Import Partner Users task
    • Self Request User Roles task
    • Approve User and Role Provisioning Requests task
    • Assign User Roles task
    • Manage Supplier User Roles and User Role Usages task
    • Map and Synchronize User Account Details task
    • Tasks for viewing account details for self or others
    • Tasks for applying and managing various role provisioning rules
    • Tasks for running synchronization processes
• Security implementation and ongoing maintenance after setup (Manage IT Security activity)
  • Implement Function Security Controls
    • Create Job Role task
    • Import Worker Users task
• Import Partner Users task
• Manage Duties task
• Manage Job Roles task
• Manage Users task
• Implement Data Security Controls
  • Manage Data Security Policies task
  • Manage Role Templates task
  • Manage Segment Security task
  • Manage Data Access Sets task
  • Define Security Profiles task group
• Auditing tasks
  • Manage Security Audit, Compliance and Reporting activity
  • Manage Application Access Controls task

Note
Go live deployment does not require lockdown or specific security tasks because security is enforced across the test to production information life cycle.

Required Roles
The following enterprise roles are provisioned to a single super user that is set up by the Oracle Fusion Applications installation process, and to the initial user set up by Oracle for Oracle Cloud Application Services:
  • Application Implementation Consultant
  • IT Security Manager
  • Application Administrators for the provisioned products
Initial security administration also includes provisioning the IT Security Manager role with Oracle Identity Management (OIM) roles for user and role management.
  • Identity User Administrator
  • Role Administrator
Additionally, the Xellerate Users organization must be assigned to the IT Security Manager role.

Important
As a security guideline, provision a dedicated security professional with the IT Security Manager role at the beginning of an implementation, and revoke that role from users provisioned with the Application Implementation Consultant role.

Tools Used to Perform Security Tasks
Security tasks are supported by tools within both Oracle Fusion Applications and Oracle Fusion Middleware.
The figure lists the tasks associated with each of the integrated products and pillars of an Oracle Fusion Applications deployment.
Security Tasks: Overview

Security tasks span multiple business processes and are performed by various roles using numerous integrated tools. The following table shows the business process model (BPM) tasks and tools used to support securing Oracle Fusion Applications.

<table>
<thead>
<tr>
<th>Example Task</th>
<th>Oracle BPM Task</th>
<th>Supporting Tools</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>View duty roles inherited by a job role</td>
<td>Manage Duties</td>
<td>• Authorization Policy Manager (APM)</td>
<td>Each logical partition or pillar contains a collection of application roles, and function and data security policies.</td>
</tr>
<tr>
<td>View entitlement or policies carried by a job role</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>In LDAP, the policy store stores application roles and the identity store stores enterprise roles.</td>
</tr>
<tr>
<td>Add a job role to a role hierarchy</td>
<td>Manage Job Roles</td>
<td>• Oracle Identity Management (OIM)</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Add a duty role to a role hierarchy</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>LDAP stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions.</td>
</tr>
<tr>
<td>Task Description</td>
<td>Module</td>
<td>Application</td>
<td>Additional Notes</td>
</tr>
<tr>
<td>--------------------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create a hierarchy of enterprise (abstract, job, data) roles</td>
<td>Manage Job Roles</td>
<td>OIM</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Create a hierarchy of (application) duty roles</td>
<td>Manage Duties</td>
<td>APM</td>
<td>The policy store stores duty roles. The identity store stores enterprise roles. Some duty roles may enable actions and their associated users interface features that your enterprise does not want users to perform in Oracle Fusion applications.</td>
</tr>
<tr>
<td>Create a new job role</td>
<td>Manage Job Roles</td>
<td>OIM</td>
<td>All functions and actions in Oracle Fusion Applications that need to be secured are covered by the reference implementation. In some cases, especially with function customizations, a new duty role may be needed.</td>
</tr>
<tr>
<td>Change duty roles inherited by a job or abstract role</td>
<td>Manage Duties</td>
<td>APM</td>
<td>The Security Reference Manuals (SRM) document the segregation of duties (SOD) policies respected within each job role.</td>
</tr>
<tr>
<td>View Segregation of Duties (SOD) policies respected by a duty role</td>
<td>Manage Application Access Controls</td>
<td>Application Access Controls Governor (AACG) in Oracle Enterprise Governance, Risk and Compliance (GRC)</td>
<td>The SRM documents the SOD conflicts for each job role.</td>
</tr>
<tr>
<td>View SOD policy violations carried by the duty roles inherited by a job role</td>
<td>Manage Application Access Controls</td>
<td>AACG in GRC</td>
<td>The Security Reference Manuals (SRM) document the SOD policies respected within each job role.</td>
</tr>
<tr>
<td>View SOD policy violations</td>
<td>Manage Segregation of Duties Policies</td>
<td>AACG in GRC</td>
<td>The SRM documents the SOD conflicts for each job role.</td>
</tr>
<tr>
<td>View the data security policies carried by a job, abstract, and data roles</td>
<td>Manage Data Security Policies</td>
<td>APM</td>
<td>Oracle Fusion Data Security stores data security policies in the policy store.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>Task Description</td>
<td>Task Details</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create and update HCM security profiles</td>
<td>Manage Data Role and Security Profiles • Oracle Fusion HCM This task does not include assigning data roles to the users, which is supported by user provisioning tasks.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create (generate) a data role</td>
<td>1. Manage Role Templates 2. Manage Data Roles and Security Profiles • APM • Oracle Fusion HCM Data roles are generated automatically based on data role templates and enterprise setup. Changes to data role templates generate new or changed data roles. Create data roles in HCM using the Manage Data Roles and Security Profiles task.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a new data security policy (not through generated data roles based on data role templates or HCM security profiles)</td>
<td>Manage Data Security Policies • APM Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View data role templates defined by a product</td>
<td>Manage Role Templates • APM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create or edit an existing data role template</td>
<td>Manage Role Templates • APM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secure common objects such as attachment categories or profile options</td>
<td>Manage Data Security Policies • APM Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View, create, update Data Access Sets used to secure Ledgers and Ledger Sets</td>
<td>Manage Data Access Sets • Oracle Fusion General Ledger</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View, create, update accounting flexfield segment security rules</td>
<td>Manage Security Segments • Oracle Fusion General Ledger</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View or update the set of job roles that can be provisioned to supplier users</td>
<td>Manage Supplier User Role • Supplier Portal • Sourcing These tools are in the Oracle Fusion Procurement product family</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Determine the supplier job roles that the supplier self service administrator can provision to supplier users</td>
<td>Manage Supplier User Role Usages • Supplier Portal • Sourcing These tools are in the Oracle Fusion Procurement product family</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set default supplier job roles based on the set of supplier roles that are defined by performing the Manage Supplier User Roles task</td>
<td>Manage Supplier User Role Usages • Supplier Portal • Sourcing These tools are in the Oracle Fusion Procurement product family</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a new implementation user</td>
<td>Create Implementation Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>Import legacy users</td>
<td>• Import Worker Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Import Partner Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a new user</td>
<td>Manage Users</td>
<td>• HCM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provision roles to a user</td>
<td>1. Provision Roles to</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Implementation Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Manage Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Oracle Fusion HCM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Oracle Sales Cloud</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Oracle Fusion Suppliers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Implementation users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>are provisioned through</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OIM since HCM is not</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>setup at the start of the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>implementation. The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Provision Roles</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>to Implementation Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>is not needed once</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>implementation is complete.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Once HCM is setup,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCM is used to provision</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>roles to non-implementation</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>users by performing the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Manage Users task.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Human Resources (HR)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>transaction flows such as</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hire and Promote also</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>provision roles.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Once supplier users are</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>setup, Supplier Model can</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>be used by internal users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>to maintain supplier user</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>accounts or supplier users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>can maintain their accounts</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>in Supplier Portal.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
View the job, abstract, and data roles provisioned to a user

<table>
<thead>
<tr>
<th>1. Manage Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Manage User Principal</td>
</tr>
<tr>
<td>3. Provision Roles to Implementation Users</td>
</tr>
</tbody>
</table>

- Human Capital Management (HCM)
- OIM

LDAP stores users, roles and provisioning information.
The Manage User Principal and Provision Roles to Implementation Users tasks are not needed once implementation is complete.

Revoke role from user.

<table>
<thead>
<tr>
<th>Manage Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCM</td>
</tr>
</tbody>
</table>

You can revoke roles from various Human Resources task flows, the HCM Manage Users task and OIM. User termination includes role revocation.

Approve role provisioning or user account request.

<table>
<thead>
<tr>
<th>Approve User and Role Provisioning Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>OIM</td>
</tr>
</tbody>
</table>

View audit logs

<table>
<thead>
<tr>
<th>Not applicable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Enterprise Manager</td>
</tr>
</tbody>
</table>

Viewing audit logs is a Oracle Fusion Middleware function and not represented by an Oracle Fusion Applications BPM task.

For more information about provisioning identities and configuring audit policies, see the Oracle Fusion Applications Administrator's Guide.

There may be more than one navigation path to the graphical user interface in which the task is performed. You can access most security tasks by starting in the Setup and Maintenance Overview page and searching for security tasks and task lists.

**Define Data Security**

**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.
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

<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>
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.
After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Data Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Manage Data Access Sets
- Manage Segment Security
- Manage Role Templates
- Manage Data Security Policies

These tasks address data security administration. For information on using the user interface pages for setting up and managing data security, see the Oracle Fusion Middleware Administrator's Guide for Authorization Policy Manager.

**Note**

The Manage Data Role and Security Profiles task, and all other HCM security profile setup tasks are documented in Human Capital Management (HCM) documentation.

**Manage Data Access Sets**

Data access sets define a set of access privileges to one or more ledgers or ledger sets.

The information on ledgers that are attached to data access sets are secured by function security. Users must have access to the segment values associated with the data access sets to access the corresponding GL account.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Data Access Administration Duty role, which is entitled to manage data access sets (the entitlement is Define General Ledger Data Access Set). This entitlement provides the access necessary to perform the Manage Data Access Sets task in General Ledger.

**Manage Segment Security**

Balancing or management segment values can secure data within a ledger.

Segment values are stored in GL_ACCESS_SET_ASSIGNMENTS and secured by restrictions, such as Exclude, on parameters that control the set of values that a user can use during data entry.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Key Flexfield Administration Duty role, which is entitled to manage application key flexfields (the entitlement is Manage
Application Key Flexfield). This entitlement provides the access necessary to perform the Manage Segment Security task in General Ledger.

**Manage Role Templates**

Data role templates automatically create or update data roles based on dimensions such as business unit. As an enterprise expands, data role templates trigger replication of roles for added dimensions. For example, when creating a new business unit, a data role template generates a new Accounts Payables Manager data role based on the Financials Common Module Template for Business Unit Security data role template.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage data role templates (the entitlement is Manage Role Template). This entitlement provides the access necessary to perform the Manage Role Templates task in APM.

**Manage Data Security Policies**

Data security grants provisioned to roles are data security policies. The security reference implementation provides a comprehensive set of predefined data security policies and predetermined data security policies based on data role templates.

Data security policies are available for review in Authorization Policy Manager (APM). Data security policies are implemented by grants stored in Oracle Fusion Data Security (FND_GRANTS).

Data security policies secure the database resources of an enterprise. Database resources are predefined applications data objects and should not be changed. However, for cases where custom database resources must be secured objects, the IT security manager is entitled to manage database resources and create new data security policies.

**Warning**

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage data security policies (the entitlement is Manage Data Security Policy). This entitlement provides the access necessary to perform the Manage Data Security Policies task in APM.

**Data Security in the Security Reference Implementation: Explained**

The reference implementation contains a set of data security policies that can be inspected and confirmed to be suitable or a basis for further implementation using the Authorization Policy Manager (APM).
The security implementation of an enterprise is likely a subset of the reference implementation, with the enterprise specifics of duty roles, data security policies, and HCM security profiles provided by the enterprise.

The business objects registered as secure in the reference implementation are database tables and views.

Granting or revoking object entitlement to a particular user or group of users on an object instance or set of instances extends the base Oracle Fusion Applications security reference implementation without requiring customization of the applications that access the data.

**Data Security Policies in the Security Reference Implementation**

The data security policies in the reference implementation entitle the grantee (a role) to access instance sets of data based on SQL predicates in a WHERE clause.

**Tip**

When extending the reference implementation with additional data security policies, identify instance sets of data representing the business objects that need to be secured, rather than specific instances or all instances of the business objects.

Predefined data security policies are stored in the data security policy store, managed in the Authorization Policy Manager (APM), and described in the Oracle Fusion Applications Security Reference Manual for each offering. A data security policy for a duty role describes an entitlement granted to any job role that includes that duty role.

**Warning**

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

The reference implementation only enforces a portion of the data security policies in business intelligence that is considered most critical to risk management without negatively affecting performance. For performance reasons it is not practical to secure every level in every dimension. Your enterprise may have a different risk tolerance than assumed by the security reference implementation.

**HCM Security Profiles in the Security Reference Implementation**

The security reference implementation includes some predefined HCM security profiles for initial usability. For example, a predefined HCM security profile allows line managers to see the people that report to them.

The IT security manager uses HCM security profiles to define the sets of HCM data that can be accessed by the roles that are provisioned to users.

**Data Roles**

The security reference implementation includes no predefined data roles to ensure a fully secured initial Oracle Fusion Applications environment.
The security reference implementation includes data role templates that you can use to generate a set of data roles with entitlement to perform predefined business functions within data dimensions such as business unit. Oracle Fusion Payables invoicing and expense management are examples of predefined business functions. Accounts Payable Manager - US is a data role you might generate from a predefined data role template for payables invoicing if you set up a business unit called US.

HCM provides a mechanism for generating HCM related data roles.

## Securing Data Access: Points to Consider

Oracle Fusion Applications supports securing data through role-based access control (RBAC) by the following methods.

<table>
<thead>
<tr>
<th>Method of securing data</th>
<th>Reason</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data roles apply explicit data security policies on job and abstract roles</td>
<td>Appropriate for job and abstract roles that should only access a subset of data, as defined by the data role template that generates the data role or by HCM security profiles.</td>
<td>Accounts Payable Manager - US data role to provide an accounts payable manager in the US business unit with access to invoices in the US business unit.</td>
</tr>
<tr>
<td>Data security policies</td>
<td>Define data access for application roles and provide inheriting job and abstract roles with implicit data security</td>
<td>Projects</td>
</tr>
</tbody>
</table>

If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

When a runtime session is created, Oracle Platform Security Services (OPSS) propagates only the necessary user to role mapping based on Oracle Fusion Data Security grants. A grant can specify entitlement to the following.

- Specific rows of data (data object) identified by primary key
- Groups of data (instance set) based on a predicate that names a particular parameter
- Data objects or instance sets based on runtime user session variables

Data is either identified by the primary key value of the row in the table where the data is stored. Or data is identified by a rule (SQL predicate) applied to the **WHERE** clause of a query against the table where the data is stored.

### Grants

Oracle Fusion Data Security can be used to restrict the following.

- Rows that are returned by a given query based on the intended business operation
- Actions that are available for a given row

Grants control which data a user can access.
Note

Attribute level security using grants requires a data security policy to secure the attribute and the entitlement check enforces that policy.

A grant logically joins a user or role and an entitlement with a static or parameterized object instance set. For example, \texttt{REGION='WEST'} is a static object instance set and \texttt{REGION=\$\texttt{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 \texttt{FND\_GRANTS} and object instance sets are stored in \texttt{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.

Manage Data Security Policies

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.

Manage Role Templates

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.
Manage Data Role and Security Profiles

HCM Data Roles: Explained

HCM data roles, like all Oracle Fusion Applications data roles, define data security policies: they enable users to perform a set of tasks, using identified menus, menu items, and pages in application user interfaces, on a specified set of data within those user interfaces. Because data roles are specific to the enterprise, no predefined HCM data roles exist.

How HCM Data Roles Differ from Other Data Roles

HCM data roles differ from other data roles in the following ways:

- You create and maintain HCM data roles outside Oracle Identity Management (OIM) and the Oracle Fusion Middleware Authorization Policy Manager (APM), and they are not based on data role templates. Although HCM data roles are visible in the Oracle Fusion Middleware APM, they must not be maintained there.

- A single HCM data role can enable access to data of multiple types.

You identify the data that users can access in HCM security profiles. You can create security profiles for the person, organization, position, country, legislative data group (LDG), document type, payroll, and payroll flow objects.

Selecting the Job Role

Each HCM data role is associated with a single job role, which you select from the list of enterprise roles. The HCM securing objects that the selected role needs to access are identified automatically, and the appropriate types of security profile are displayed. For example, if you select the job role human resource analyst, users with that job role need to access managed person, public person, organization, position, LDG, and document type data; therefore, security profiles for those object types must be included in the HCM data role. The security profile types that appear in the HCM data role vary according to the data requirements of the selected job role.

If you select a job role that requires no access to HCM data secured by security profiles, you cannot create an HCM data role.

Note

If you create custom job roles in OIM, you must add them to a locally defined role category that ends with “Job Roles”; otherwise, they do not appear in the list of job roles when you create an HCM data role. Do not add custom job roles to the predefined role category HCM - Job Roles.

Creating or Selecting the Security Profiles

You can either create new security profiles or use existing security profiles. For each object type, you can include only one security profile in an HCM data role.
Users with Multiple HCM Data Roles

When users have multiple HCM data roles, the data security policies arising from each role remain separate. For example, being able to promote or terminate workers in the purchasing department in one HCM data role and view contact details of all workers in the sales department in another HCM data role does not enable a user to promote or terminate workers in the sales department.

Components of the HCM Data Role

The following figure summarizes how the components of the HCM data role contribute to Oracle Fusion Data Security for the data role. Oracle Fusion Data Security comprises the data security policies for data roles that are generated automatically when data roles are created.

The job role that you select in the HCM data role inherits multiple duty roles. Each duty role has one or more function privileges and related data privileges, from which the relevant HCM objects are identified. The specific instances of the objects required by this HCM data role are identified in security profiles and stored in a data instance set. Data security policy data is created automatically in Oracle Fusion Data Security when you create the data role.

For example, the human resource specialist job role inherits the employee hire and worker promotion duty roles, among many others. The inherited duty roles provide both function privileges, such as Hire Employee, Rehire Employee, and Promote Workers, and data privileges to HCM objects, such as person and assignment. The specific instances of those objects required by this HCM data role, such as people with assignments in a specified legal employer and department, are identified in security profiles.
HCM Security Profiles: Explained

A security profile defines the criteria that identify instances of a human capital management (HCM) object. For example, a person security profile defines the criteria that identify one or more person records, and a position security profile defines the criteria that identify one or more positions. When you include a security profile in an HCM data role and provision the data role to a user, that user can access the data instances identified in the security profile. The type of access available to the user (for example whether the user can edit or simply view the data) depends on the job role identified in the HCM data role.

HCM Object Types

You can create security profiles for the following HCM object types:

- Person
- Managed person
- Public person
- Organization
- Position
- Legislative data group (LDG)
- Country
- Document type
- Payroll
- Payroll flow

All security profile definitions for these HCM objects are eventually visible in the Oracle Fusion Middleware Authorization Policy Manager (APM). The name of the security profile's data instance set in the Oracle Fusion Middleware APM is derived from the name of the security profile and the relevant object type. For example, if the security profile name is Manager Hierarchy, then the data instance set for the object PER_ALL_PEOPLE_F is HCM:PER:PER_ALL_PEOPLE_F:Manager Hierarchy.

You must use the Oracle Fusion Human Capital Management interfaces, which are designed for ease of use and access, to create and maintain security profiles; do not use the Oracle Fusion Middleware APM to maintain security profiles for these HCM objects.

Security Criteria in HCM Security Profiles

In any HCM security profile, you specify the criteria that identify data instances of the relevant type. For example, in an organization security profile, you can identify organizations by organization hierarchy, by organization classification, or by listing organizations to include in or exclude from the security profile. All of the criteria in an HCM security profile apply when the data instance set is defined; for example, if you identify organizations by both organization hierarchy and organization classification, then both sets of criteria apply, and only those organizations that satisfy all criteria belong to the data instance set.

Predefined HCM Security Profiles

The following HCM security profiles are predefined:
### Security Profile Name | HCM Security Profile Type | Description |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>View All People</td>
<td>Person</td>
<td>Identifies all person records in the enterprise</td>
</tr>
<tr>
<td>View Own Record</td>
<td>Person</td>
<td>Identifies the signed-on user's own person record and the person records of that user's contacts</td>
</tr>
<tr>
<td>View Manager Hierarchy</td>
<td>Person</td>
<td>Identifies the signed-on user's line manager hierarchy</td>
</tr>
<tr>
<td>View All Workers</td>
<td>Person</td>
<td>Identifies the person records of all people who have a work relationship in the enterprise</td>
</tr>
<tr>
<td>View All Organizations</td>
<td>Organization</td>
<td>Identifies all organizations in the enterprise</td>
</tr>
<tr>
<td>View All Positions</td>
<td>Position</td>
<td>Identifies all positions in the enterprise</td>
</tr>
<tr>
<td>View All Legislative Data Groups</td>
<td>LDG</td>
<td>Identifies all LDGs in the enterprise</td>
</tr>
<tr>
<td>View All Countries</td>
<td>Country</td>
<td>Identifies all countries in the FND_TERRITORIES table</td>
</tr>
<tr>
<td>View All Document Types</td>
<td>Document Type</td>
<td>Identifies all document types in the enterprise</td>
</tr>
<tr>
<td>View All Payrolls</td>
<td>Payroll</td>
<td>Identifies all payrolls in the enterprise</td>
</tr>
<tr>
<td>View All Flows</td>
<td>Payroll Flow</td>
<td>Identifies all payroll flows in the enterprise</td>
</tr>
</tbody>
</table>

You can include the predefined security profiles in any HCM data role, but you cannot edit them. Note also that the View all option is disabled in any security profile that you create; this restriction exists because predefined security profiles exist for this requirement.

### Creating Security Profiles

You can create security profiles either individually or as part of the process of creating an HCM data role. If you have standard requirements, it may be more efficient to create the security profiles individually and include them in appropriate HCM data roles.

### Reusability and Inheritance of Security Profiles

Regardless of how you create them, all security profiles are reusable; they do not belong to particular HCM data roles, and you can include them in any HCM data role for which they define an appropriate data instance set.

You can include security profiles in other security profiles. For example, you can include an organization security profile:

- In a person security profile, to secure person records by department, business unit, or legal employer
- In a position security profile, to secure positions by department or business unit

Therefore, one security profile can inherit the data instance set defined by another.
Assigning Security Profiles to Abstract Roles: Explained

Abstract roles define a worker’s role in the enterprise independently of the job that the worker is hired to do.

These abstract roles are predefined in Oracle Fusion Human Capital Management:

- Line manager
- Employee
- Contingent worker

Enabling Data Access for Abstract Roles

Typically, you create role mappings during implementation to provision abstract roles automatically to eligible workers. Although users with these roles may be able to sign in to Oracle Fusion Applications and navigate to tasks of interest, they have no automatic access to data. For example, employees can navigate to the Person Gallery but cannot view portraits or see lists of person names in product interfaces, and line managers can navigate to the Manager Resources Dashboard but can see no data for their organizations. To enable users with abstract roles to access relevant HCM data, you must assign security profiles to those abstract roles.

Predefined Security Profiles to Assign to Abstract Roles

To enable users with abstract roles to access relevant data, you assign the following predefined security profiles directly to the employee, contingent worker, and line manager abstract roles.

<table>
<thead>
<tr>
<th>Security Profile Type</th>
<th>Employee</th>
<th>Contingent Worker</th>
<th>Line Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person</td>
<td>View Own Record</td>
<td>View Own Record</td>
<td>View Manager Hierarchy</td>
</tr>
<tr>
<td>Public person</td>
<td>View All Workers</td>
<td>View All Workers</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Organization</td>
<td>View All Organizations</td>
<td>View All Organizations</td>
<td>View All Organizations</td>
</tr>
<tr>
<td>Position</td>
<td>View All Positions</td>
<td>View All Positions</td>
<td>View All Positions</td>
</tr>
<tr>
<td>Legislative data group</td>
<td>View All Legislative Data Groups</td>
<td>View All Legislative Data Groups</td>
<td>View All Legislative Data Groups</td>
</tr>
<tr>
<td>Country</td>
<td>View All Countries</td>
<td>View All Countries</td>
<td>View All Countries</td>
</tr>
<tr>
<td>Document type</td>
<td>View All Document Types</td>
<td>View All Document Types</td>
<td>View All Document Types</td>
</tr>
<tr>
<td>Payroll Flow</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>View All Flows</td>
</tr>
</tbody>
</table>

After implementation, you may want to change aspects of this data access. For example, you may want to create your own security profiles and assign those directly to abstract roles; however, you must remember that such changes apply to all users who have the abstract role.

HCM Data Roles

Users who have abstract roles are likely to gain additional data access by means of HCM data roles that you define for their job roles. For example, you may create an HCM data role for human resource specialists to enable them to access...
the person records of all workers in a legal employer. Such data access is in addition to any data access provided by abstract roles.

Assigning Security Profiles to Abstract Roles: Worked Example

This example shows how to assign predefined security profiles to the employee, contingent worker, and line manager abstract roles.

Searching for the Employee Abstract Role

1. On the All Tasks tab of the Overview page of the Setup and Maintenance work area, search for the task Manage Data Role and Security Profiles.
2. In the Search Results region, click Go to Task.
3. On the Manage Data Roles and Security Profiles page, enter the abstract-role name Employee in the Role field. Click Search.
4. In the Search Results region, highlight the entry for the predefined Employee role and click Assign.

Assigning Security Profiles to the Employee Abstract Role

1. On the Assign Data Role: Security Criteria page, select the security-profile values shown in the following table. These are the security profiles that are typically assigned to the employee role. You may see a subset of these security profiles, depending on the combination of product offerings that you are implementing.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization Security Profile</td>
<td>View All Organizations</td>
</tr>
<tr>
<td>Position Security Profile</td>
<td>View All Positions</td>
</tr>
<tr>
<td>Country Security Profile</td>
<td>View All Countries</td>
</tr>
<tr>
<td>LDG Security Profile</td>
<td>View All Legislative Data Groups</td>
</tr>
<tr>
<td>Person Security Profile (Person section)</td>
<td>View Own Record</td>
</tr>
<tr>
<td>Person Security Profile (Public Person section)</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Document Type Security Profile</td>
<td>View All Document Types</td>
</tr>
</tbody>
</table>

2. Click Review.
3. On the Assign Data Role: Review page, click Submit.
4. On the Manage Data Roles and Security Profiles page, search again for the predefined Employee role.
5. In the Search Results region, confirm that a green check mark appears in the Security Profiles column for the Employee role. The check mark confirms that security profiles are assigned to the role.

Repeat the steps in Searching for the Employee Abstract Role and Assigning Security Profiles to the Employee Abstract Role for the predefined Contingent Worker role.
Searching for the Line Manager Abstract Role

1. On the Manage Data Roles and Security Profiles page, enter the abstract-role name Line Manager in the Role field. Click Search.

2. In the Search Results region, highlight the entry for the predefined Line Manager role and click Assign.

Assigning Security Profiles to the Line Manager Abstract Role

1. On the Assign Data Role: Security Criteria page, select the security-profile values shown in the following table. These are the security profiles that are typically assigned to the line manager role. You may see a subset of these security profiles, depending on the combination of product offerings that you are implementing.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization Security Profile</td>
<td>View All Organizations</td>
</tr>
<tr>
<td>Position Security Profile</td>
<td>View All Positions</td>
</tr>
<tr>
<td>LDG Security Profile</td>
<td>View All Legislative Data Groups</td>
</tr>
<tr>
<td>Person Security Profile (Person section)</td>
<td>View Manager Hierarchy</td>
</tr>
<tr>
<td>Person Security Profile (Public Person section)</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Document Type Security Profile</td>
<td>View All Document Types</td>
</tr>
<tr>
<td>Payroll Flow</td>
<td>View All Flows</td>
</tr>
</tbody>
</table>

2. Click Review.

3. On the Assign Data Role: Review page, click Submit

4. On the Manage Data Roles and Security Profiles page, search again for the predefined Line Manager role.

5. In the search results, confirm that a green check mark appears in the Security Profiles column for the Line Manager role. The check mark confirms that security profiles are assigned to the role.

Define Users

Securing Identities and Users: Points To Consider

Identity covers all aspects of an entity’s existence within the contexts in which it is used. The identity of an enterprise user consists of HR attributes, roles, resources, and relationships.

HR attributes include identifying information about a user that is relatively static and well understood, such as first and last name, title, and job function.
Roles are part of a user's identity and define the user's purpose and responsibilities.

Within identity management, resources define what a user can and does do. In an enterprise, this typically translates into what resources a user has access to, what privileges they have on that resource, and what they have been doing on that resource. Resources can be application accounts or physical devices such as laptops or access cards. The enterprise owns the resources, secures them, and manages access to the resources by managing the user's identity and access.

Relationships establish the portion of user identities that involve organizational transactions such as approvals.

An Oracle Fusion Applications user and corresponding identity are usually created in a single transaction, such as when a worker is created in Human Resources (HR). That transaction automatically triggers provisioning requests for the user based on role provisioning rules.

User accounts for some identities that are not employees, such as partner contacts, may be created in a later transaction using an identity that is already created in the identity store. Supplier contacts are created in the Supplier Model, not HR.

**Stores**

Various locations store identity and user data.

Identity data consists of the following.

- HR person records
- Oracle Fusion Trading Community Model party records

In Oracle Fusion Applications, identities and users correspond one to one, but not all identities correspond to a user, and not all users are provisioned with an identity. Some identities stored in HR and Trading Community Model may not be provisioned to user accounts and therefore are not synchronized with Oracle Identity Management (OIM). For example, a contact for a prospective customer is an identity in Trading Community Model but may not be provisioned with a user account in OIM. Some users stored in the Lightweight Directory Access Protocol (LDAP) store may not be provisioned with identities. For example, system user accounts used to run Web services to integrate third party services with Oracle Fusion Applications are not associated with a person record in HR or Trading Community Model. Some identifying credentials such as name, department, e-mail address, manager, and location are stored with user data in the LDAP store.

**Importing Users**

You can import users or user attributes in bulk from existing legacy identity and user stores.

Your tasks may include the following.

- Create users in bulk
- Update specific attributes for all users, such as postal code
- Link users to HR or Trading Community Model persons
• Monitor progress of the import process
• Correct errors & re-import
• Export users in bulk
• Import and export users using a standard plain text data interchange format like Lightweight Data Interchange Format (LDIF)

You can reserve a specific user name not currently in use for use in the future, or release a reserved username from the reservation list and make it available for use. Between a user registration request and approved registration, Oracle Fusion Applications holds the requested user name on the reservation list, and releases the name if an error occurs in the self-registration process or the request is rejected. Self-registration processes check the reservation list for user name availability and suggest alternative names.

Provisioning Events

New identities, such as new hires, trigger user and role provisioning events. In addition to user creation tasks, other tasks, such as Promote Worker or Transfer Worker, result in role provisioning and recalculation based on role provisioning rules.

When an identity’s attributes change, you may need to provision the user with different roles. Role assignments may be based on job codes, and a promotion triggers role provisioning changes. Even if the change in the identities attributes requires no role assignment change, such as with a name change, OIM synchronizes the corresponding user information in the LDAP store.

Deactivating or terminating an identity triggers revocation of some roles to end all assignments, but may provision new roles needed for activities, such as a pay stub review. If the corresponding user for the identity was provisioned with a buyer role, terminating the identity causes the user’s buyer record in Procurement to be disabled, just as the record was created when the user was first provisioned with the buyer role.

Notifications and Audits

Oracle Fusion Applications provides mechanisms for notifying and auditing requests or changes affecting identities and users.

Oracle Fusion Applications notifies requestors, approvers, and beneficiaries when a user account or role is provisioned. For example, when an anonymous user registers as a business-to-customer (B2C) user, the B2C user must be notified of the registration activation steps, user account, password and so on once the approver (if applicable) has approved the request and the user is registered in the system.

User ID and GUID attributes are available in Oracle Fusion Applications session information for retrieving authenticated user and identity data.

End user auditing data is stored in database WHO columns and used for the following activities.
• Setting up sign-in audit
• Using the application monitor
• Notifying of unsuccessful sign ins
• Sign-in audit reports

You can conduct real time audits that instantiate a runtime session and impersonate the target user (with the proxy feature) to test what a user has access to under various conditions such as inside or outside firewall and authentication level.

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

Delegated Administration

You can designate local administrators as delegated administrators to manage a subset of users and roles.

Delegated administrators can be internal or external persons who are provisioned with a role that authorizes them to handle provisioning events for a subset of users and roles.

For example, internal delegated administrators could be designated to manage users and roles at the division or department level. External delegated administrators could be designated to manage users and roles in an external organization such as a primary supplier contact managing secondary users within that supplier organization.

You can also define delegated administration policies based on roles. You authorize users provisioned with specific roles named in the policy to request a subset of roles for themselves if needed, such as authorizing a subset of roles for a subset of people. For example, the policy permits a manager of an Accounts Payables department to approve a check run administrator role for one of their subordinates, but prohibits the delegated administrator from provisioning a budget approver role to the subordinate.

Credentials

You activate or change credentials on users by managing them in Oracle Identity Management (OIM)

Applications themselves must be credentialed to access one another.

Oracle Fusion Applications distinguishes between user identities and application identities (APPID). Predefined application identities serve to authorize jobs and transactions that require higher privileges than users.

For example, a payroll manager may submit a payroll run. The payroll application may need access to the employee's taxpayer ID to print the payslip. However, the payroll manager is not authorized to view taxpayer IDs in the user interface as they are considered personally identifiable information (PII).

Calling applications use application identities (APPID) to enable the flow of transaction control as it moves across trust boundaries. For example, a user in the Distributed Order Orchestration product may release an order for shipping. The code that runs the Pick Notes is in a different policy store than the code
that releases the product for shipment. When the pick note printing program is invoked it is the Oracle Fusion Distributed Order Orchestration Application Development Framework (ADF) that is invoking the program and not the end user.

Manage HCM Role Provisioning Rules

Role Provisioning and Deprovisioning: Explained

A user’s access to data and functions depends on the user’s roles: users have one or more roles that enable them to perform the tasks required by their jobs or positions. Roles must be provisioned to users; otherwise, users have no access to data or functions.

Role Provisioning Methods

Roles can be provisioned to users:

- Automatically
- Manually, using delegated administration:
  - Users such as line managers and human resource specialists can provision roles manually to other users.
  - Users can request roles for themselves.

For both automatic and manual role provisioning, you create a role mapping to identify when a user becomes eligible for a role.

Oracle Identity Management (OIM) can be configured to notify users when their roles change; notifications are not issued by default.

Role Types

Data roles, abstract roles, and job roles can be provisioned to users. Roles available for provisioning include predefined roles, HCM data roles, and roles created using OIM.

Automatic Role Provisioning

A role is provisioned to a user automatically when at least one of the user’s assignments satisfies the conditions specified in the relevant role-mapping definition. The provisioning occurs when the assignment is either created or updated. For example, when a person is promoted to a management position, the line manager role is provisioned automatically to the person if an appropriate role mapping exists. Any change to a person’s assignment causes the person’s automatically provisioned roles to be reviewed and updated as necessary.

Role Deprovisioning

Automatically provisioned roles are deprovisioned automatically as soon as a user no longer satisfies the role-mapping conditions. For example, a line
manager role that is provisioned to a user automatically is deprovisioned automatically when the user ceases to be a line manager.

Automatically provisioned roles can be deprovisioned manually at any time.

Manually provisioned roles are deprovisioned automatically only when all of the user’s work relationships are terminated; in all other circumstances, users retain manually provisioned roles until they are deprovisioned manually.

**Changes to Assignment Managers**

When a person’s line manager is changed, the roles of both new and previous line managers are updated as necessary. For example, if the person’s new line manager now satisfies the conditions in the role mapping for the line manager role, and the role is one that is eligible for autoprovisioning, then that role is provisioned automatically to the new line manager. Similarly, if the previous line manager no longer satisfies the conditions for the line manager role, then that role is deprovisioned automatically.

**Roles at Termination**

When a work relationship is terminated, all automatically provisioned roles for which the user does not qualify in other work relationships are deprovisioned automatically. Manually provisioned roles are deprovisioned automatically only if the user has no other work relationships; otherwise, the user retains all manually provisioned roles until they are deprovisioned manually.

Automatic deprovisioning can occur either as soon as the termination is submitted or approved or on the day after the termination date. The user who is terminating the work relationship selects the appropriate deprovisioning date.

Role mappings can provision roles to users automatically at termination. For example, the locally defined roles Retiree and Beneficiary could be provisioned to users at termination based on assignment status and person type values.

If a termination is later reversed, roles that were deprovisioned automatically at termination are reinstated and post-termination roles are deprovisioned automatically.

**Date-Effective Changes to Assignments**

Automatic role provisioning and deprovisioning are based on current data. For a future-dated transaction, such as a future promotion, role changes are identified and role provisioning occurs on the day the changes take effect, not when the change is entered. The process Send Pending LDAP Requests identifies future-dated transactions and manages role provisioning and deprovisioning at the appropriate time. Note that such role-provisioning changes are effective as of the system date; therefore, a delay of up to 24 hours may occur before users in other time zones acquire the access for which they now qualify.

**Role Mappings: Explained**

User access to data and functions is determined by abstract, job, and data roles, which are provisioned to users either automatically or manually. To enable a role to be provisioned to users, you define a relationship, known as a mapping, between the role and a set of conditions, typically assignment attributes such as department, job, and system person type. In a role mapping, you can select
any role stored in the Lightweight Directory Access Protocol (LDAP) directory, including Oracle Fusion Applications predefined roles, roles created in Oracle Identity Management (OIM), and HCM data roles.

The role mapping can support:

- Automatic provisioning of roles to users
- Manual provisioning of roles to users
- Role requests from users
- Immediate provisioning of roles

**Automatic Provisioning of Roles to Users**

A role is provisioned to a user automatically if:

- At least one of the user's assignments satisfies all conditions associated with the role in the role mapping.
- You select the Autoprovision option for the role in the role mapping.

For example, for the HCM data role Sales Manager Finance Department, you could select the Autoprovision option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>Finance Department</td>
</tr>
<tr>
<td>Job</td>
<td>Sales Manager</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

The HCM data role Sales Manager Finance Department is provisioned automatically to users with at least one assignment that satisfies all of these conditions.

Automatic role provisioning occurs as soon as the user is confirmed to satisfy the role-mapping conditions, which can be when the user's assignment is either created or updated. The provisioning process also removes automatically provisioned roles from users who no longer satisfy the role-mapping conditions.

**Note**

The automatic provisioning of roles to users is effectively a request to OIM to provision the role. OIM may reject the request if it violates segregation-of-duties rules or fails a custom OIM approval process.

**Manual Provisioning of Roles to Users**

Users such as human resource (HR) specialists and line managers can provision roles manually to other users; you create a role mapping to identify roles that can be provisioned in this way.

Users can provision a role to other users if:

- At least one of the assignments of the user who is provisioning the role (for example, the line manager) satisfies all conditions associated with the role mapping.
- You select the Requestable option for the role in the role mapping.
For example, for the HCM data role Quality Assurance Team Leader, you could select the **Requestable** option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies both of these conditions can provision the role Quality Assurance Team Leader manually to other users, who are typically direct and indirect reports.

If the user’s assignment subsequently changes, there is no automatic effect on roles provisioned by this user to others; they retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

**Role Requests from Users**

Users can request roles when reviewing their own account information; you create a role mapping to identify roles that users can request for themselves.

Users can request a role if:

- At least one of their own assignments satisfies all conditions associated with the role mapping.
- You select the **Self-requestable** option for the role in the role mapping.

For example, for the Expenses Reporting role you could select the **Self-requestable** option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>ABC Department</td>
</tr>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies all of these conditions can request the role. The user acquires the role either immediately or, if approval is required, once the request is approved. Self-requested roles are classified as manually provisioned.

If the user’s assignment subsequently changes, there is no automatic effect on self-requested roles. Users retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

**Immediate Provisioning of Roles**

When you create a role mapping, you can apply autoprovisioning from the role mapping itself.

In this case, all assignments and role mappings in the enterprise are reviewed. Roles are:

- Provisioned immediately to all users who do not currently have roles for which they are eligible
• Deprovisioned immediately from users who are no longer eligible for roles that they currently have

Immediate autoprovisioning from the role mapping enables bulk automatic provisioning of roles to a group of users who are identified by the role-mapping conditions. For example, if you create a new department after a merger, you can provision relevant roles to all users in the new department by applying autoprovisioning immediately.

To provision roles immediately to a single user, the user’s line manager or an HR specialist can autoprovion roles from that user’s account.

**Role-Mapping Names**

The names of role mappings must be unique in the enterprise. You are recommended to devise a naming scheme that reveals the scope of each role mapping. For example:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoprovisioned Roles Sales Department</td>
<td>Mapping includes all roles provisioned automatically to anyone in the sales department</td>
</tr>
<tr>
<td>Benefits Specialist Autoprovisioned</td>
<td>Mapping defines the conditions for autoprovisioning the Benefits Specialist role</td>
</tr>
<tr>
<td>Line Manager Requestable Roles</td>
<td>Mapping includes all roles that a line manager can provision manually to direct and indirect reports</td>
</tr>
</tbody>
</table>

**Role Mappings: Examples**

Roles must be provisioned to users explicitly, either automatically or manually; no role is provisioned to a user by default. This topic provides some examples of typical role mappings to support automatic and manual role provisioning.

**Creating a Role Mapping for Employees**

You want all employees in your enterprise to have the Employee role automatically when they are hired. In addition, employees must be able to request the Expenses Reporting role when they need to claim expenses. Few employees will need this role, so you decide not to provision it automatically to all employees.

You create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

• Employee role, and select the **Autoprovion** option
• Expenses Reporting role, and select the **Self-requestable** option
You could create a similar role mapping for contingent workers called All Contingent Workers, where you would set the system person type to contingent worker.

**Note**

If the Employee and Contingent Worker roles are provisioned automatically, pending workers acquire them when their periods of employment or placements start. If they need roles before then, you create a separate role mapping for the pending worker system person type.

### Creating a Role Mapping for Line Managers

Any type of worker can be a line manager in the sales business unit. You create a role mapping called Line Manager Sales BU and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Unit</td>
<td>Sales</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
</tbody>
</table>

You include the Line Manager role and select the **Autoprovision** option. This role mapping ensures that the Line Manager role is provisioned automatically to any worker with at least one assignment that matches the role-mapping conditions.

In the same role mapping, you could include roles that line managers in this business unit can provision manually to other users by selecting the roles and marking them as requestable. Similarly, if line managers can request roles for themselves, you could include those in the same role mapping and mark them as self-requestable.

### Creating a Role Mapping for Retirees

Retirees in your enterprise need a limited amount of system access to manage their retirement accounts. You create a role mapping called All Retirees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Retiree</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Inactive</td>
</tr>
</tbody>
</table>

You include the locally defined role Retiree in the role mapping and select the **Autoprovision** option. When at least one of a worker's assignments satisfies the role-mapping conditions, the Retiree role is provisioned to that worker automatically.

### Creating a Role Mapping for Sales Managers

Grade 6 sales managers in the sales department need the Sales Manager role. In addition, sales managers need to be able to provision the Sales Associate role to other workers. You create a role mapping called Sales Managers Sales Department and enter the following conditions.
In the role mapping, you include the:

- Sales Manager role, and select the **Autoprovision** option
- Sales Associate role, and select the **Requestable** option

### Import Worker Users

#### Defining Security After Enterprise Setup: Points to Consider

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Import Worker Users
- Import Partner Users
- Manage Job Roles
- Manage Duties
- Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human Capital Management (HCM), Supplier Portal, and Oracle Sales Cloud Self Service, respectively, and associate them with users.

#### Before Importing Users

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Initiate HCM Spreadsheet Load task. If user and role provisioning rules have been defined, the Initiate HCM Spreadsheet Load process automatically creates user and role provisioning requests as the workers are created.

Once the enterprise is set up, performing the Initiate HCM Spreadsheet Load task populates the enterprise with HR workers in records linked by global user
ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Initiate HCM Spreadsheet Load task results in new user accounts being created. Worker email addresses as an alternate input for the Initiate HCM Spreadsheet Load task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Initiate HCM Spreadsheet Load task in HCM.

Note

The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in Oracle Sales Cloud.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

Import Users

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks.

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Initiate HCM Spreadsheet Load task to create users from imported HR workers.

Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager.

In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account...
Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

**Manage Job Roles**

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

**Note**

Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, Oracle Sales Cloud, or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

**Manage Duties**

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering.

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.
In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

**Tip**

As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

**Note**

Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

**Manage Application Access Controls**

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

**Note**

In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

**Importing Worker Users: Explained**

You can import workers from legacy applications to Oracle Fusion Applications using the Import Worker Users task. By enabling you to bulk-load existing data, this task is an efficient way of creating and enabling users of Oracle Fusion Applications.

**The Import Worker Users Process**

Importing worker users is a two-stage process:
1. When you perform the Import Worker Users task, the Initiate Spreadsheet Load page opens. On the Initiate Spreadsheet Load page, you generate and complete the Create Worker spreadsheet. You must map your data to the spreadsheet columns and provide all required attributes. Once the spreadsheet is complete, you click Upload in the spreadsheet to import the data to the Load Batch Data stage tables.

2. As valid data rows are imported to the Load Batch Data stage tables, the Load Batch Data process runs automatically. Load Batch Data is a generic utility for loading data to Oracle Fusion Human Capital Management from external sources. This process loads data from the Load Batch Data stage tables to the Oracle Fusion application tables.

User-Account Creation

Oracle Fusion user accounts are created automatically for imported workers in Oracle Identity Management (OIM), unless automatic account creation is disabled.

By default, user account names and passwords are sent automatically to users when their accounts are created. This default action may have been changed at enterprise level, as follows:

- User account names and passwords may be sent to an enterprise-wide e-mail rather than to users themselves.
- Automatic sending of user account names and passwords may be disabled for the enterprise. In this case, you can notify users at an appropriate time.

Role Provisioning

Once user accounts exist, roles are provisioned to users automatically in accordance with current role-provisioning rules. For example, current rules could provision the employee abstract role to every worker. Role provisioning occurs automatically unless it is disabled for the enterprise.

Importing Worker Users: Worked Example

This example shows how to import worker users from legacy applications to Oracle Fusion Applications.

The following table summarizes key decisions for this task.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is my spreadsheet name?</td>
<td>WorkersMMDYYBatchnn.xlsx</td>
</tr>
<tr>
<td>You can define your own naming convention.</td>
<td>For example, Workers042713Batch01.xlsx.</td>
</tr>
<tr>
<td>In this example, the name is selected to</td>
<td></td>
</tr>
<tr>
<td>make identifying the spreadsheet contents</td>
<td></td>
</tr>
<tr>
<td>easy.</td>
<td></td>
</tr>
<tr>
<td>What is my batch name?</td>
<td>Workers042713Batchnn</td>
</tr>
<tr>
<td>You can define your own batch name, which</td>
<td></td>
</tr>
<tr>
<td>must be unique. In this example, the batch</td>
<td></td>
</tr>
<tr>
<td>name is the same as the spreadsheet name.</td>
<td></td>
</tr>
</tbody>
</table>
Summary of the Tasks

Import worker users by:

1. Selecting the Import Worker Users task
2. Creating the spreadsheet
3. Entering worker data in the spreadsheet
4. Importing worker data and correcting import errors
5. Reviewing and correcting load errors

Prerequisites

Before you can complete this task, you must have:

1. Installed the desktop client Oracle ADF Desktop Integration Add-in for Excel
2. Enabled the Trust Center setting Trust access to the VBA project object model in Microsoft Excel

Selecting the Import Worker Users Task

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>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Task</td>
</tr>
<tr>
<td>Name</td>
<td>Import Worker Users</td>
</tr>
</tbody>
</table>

3. Click Search.
4. In the search results, click Go to Task for the task Import Worker Users.

The Initiate Spreadsheet Load page opens.

Alternatively, you can select the Import Worker Users task from an implementation project.

Creating the Spreadsheet

1. On the Initiate Spreadsheet Load page, find the entry for Create Worker in the list of business objects.

Create Worker appears after other business objects such as departments, locations, and jobs. Those business objects must be created before worker users, regardless of how you create them.

2. Click Create Spreadsheet for the Create Worker entry.
3. When prompted, save the spreadsheet locally using the name Workers042713Batch01.xlsx.
4. When prompted, sign in to Oracle Fusion Applications using your Oracle Fusion user name and password.
Entering Worker Data in the Spreadsheet

1. In the Batch Name field of the spreadsheet Workers042713Batch01.xlsx, replace the default batch name with the batch name Workers042713Batch01.

2. If your data includes flexfields, click Configure Flexfield to configure flexfield data. Otherwise, go to step 5 of this task.

3. In the Configure Flexfield window, select an attributes value and click OK.

4. See the Flexfields Reference tab for information about the configured flexfield.

5. Enter worker data in the spreadsheet.
   Ensure that you provide any required values and follow instructions in the spreadsheet for creating rows.

Importing Worker Data and Correcting Import Errors

Use the default values except where indicated.

1. In the workers spreadsheet, click Upload.

2. In the Upload Options window, click OK.
   As each row of data is uploaded to the Load Batch Data stage tables, its status is updated.

3. When uploading completes, identify any spreadsheet rows with the status Insert Failed, which indicates that the row was not imported to the stage tables.

4. For any row that failed, double-click the status value to display a description of the error.

5. Correct any import errors and click Upload again to import the remaining rows to the same batch.
   As rows are imported successfully to the stage tables, the data is loaded automatically to the application tables.

Reviewing and Correcting Load Errors

1. In the spreadsheet, click Refresh to display latest load status.
   Any errors that occur during the load process are reported in the spreadsheet.

2. Correct any load errors in the spreadsheet.

3. Repeat this process from Importing Worker Data and Correcting Import Errors until all spreadsheet rows are both imported and loaded successfully.

4. Close the spreadsheet.
   To load a second batch of worker users on the same date, increment the batch number in the spreadsheet and batch names (for example, Workers042713Batch02).
Manage Users

Creating Users: Worked Example

You can create users by entering basic person and employment data. A user account is created automatically for a person when you create the user record. You can assign the users Oracle Fusion Human Capital Management (HCM) and non-HCM data roles, each providing access to specific functions and data. This example demonstrates how to create a user and assign roles to the user.

**Note**

This user management functionality is available for HCM Foundation and Oracle Fusion Workforce Directory Management (WDM) users only.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>For whom are you creating the user record?</td>
<td>Gail Williams</td>
</tr>
<tr>
<td>What is the user account name?</td>
<td>Same as the e-mail ID, <a href="mailto:gail.williams@vision.com">gail.williams@vision.com</a></td>
</tr>
<tr>
<td>Where is Gail employed?</td>
<td>Gail is an employee of Vision Corporation, and works in the Human Resources (HR) department in the Canada office.</td>
</tr>
<tr>
<td>What roles must be provisioned to Gail?</td>
<td>Autoprovision the employee role. Gail is responsible for processing workers’ expense claims so provision the role Expense Claims Administrator manually to Gail.</td>
</tr>
</tbody>
</table>

**Prerequisites**

1. Create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

- Employee role, and select the **Autoprovision** option
- Expense Claims Administrator role, and select the **Self-requestable** option

**Creating a User**

1. On the Search Person page, click the **Create** icon to open the Create User page.
2. Complete the fields, as shown in this table:
3. In the User Details region, leave the User Name field blank. The user name defaults to the user's e-mail ID.

4. In the Employment Information region, select the person type **Employee** and complete the fields as shown in the table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>Williams</td>
</tr>
<tr>
<td>First Name</td>
<td>Gail</td>
</tr>
<tr>
<td>E-Mail</td>
<td><a href="mailto:gail.williams@vision.com">gail.williams@vision.com</a></td>
</tr>
<tr>
<td>Hire Date</td>
<td>4/12/11</td>
</tr>
</tbody>
</table>

**Assigning Roles to the User**

1. Click **Autoprovision Roles** to provision the employee role to the user.

2. Click **Add Role**.

3. Search for and select the **Expense Claims Administrator** role.

4. Click **Save and Close**. The user account is created and the roles are assigned to the user immediately.

**User Details System Extract Report**

The Oracle BI Publisher User Details System Extract Report includes details of some or all Oracle Fusion Applications user accounts.

To run this report, you must have an HCM data role that provides view-all access to person records for the Human Capital Management Application Administrator job role.

To run the report:

1. Navigate to **Tools - Reports and Analytics**.

2. In the Contents pane of the Reports and Analytics work area, navigate to **Shared Folders - Human Capital Management - Workforce Management - Human Resources Dashboard**.

3. Select the User Details System Extract report.

4. In the report window, click **More**.

5. On the Oracle Business Intelligence page for the report, select **Open** to run the report immediately or **Schedule** to schedule the report.

**Parameters**

**User Population**
Enter one of the following values to identify the group of user accounts to include in the report.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCM</td>
<td>User accounts with an associated HCM person record.</td>
</tr>
<tr>
<td>TCA</td>
<td>User accounts with an associated TCA party account.</td>
</tr>
<tr>
<td>OIM</td>
<td>Accounts for users in the PER_USERS table who do not have an associated person number or party ID. OIM users are also referred to as implementation users.</td>
</tr>
<tr>
<td>ALL</td>
<td>HCM, TCA, and OIM users accounts.</td>
</tr>
</tbody>
</table>

**From Date**

Accounts for HCM and OIM users created on or after this date are included in the report. If you specify no **From Date** value, then accounts with any creation date are included, subject only to any **To Date** value that you specify.

From and to dates do not apply to the TCA user population; the report includes all TCA users if you include them in the report's user population.

**To Date**

Accounts for HCM and OIM users created on or before this date are included in the report. If you specify no **To Date** value, then accounts with any creation date are included, subject only to any **From Date** value that you specify.

From and to dates do not apply to the TCA user population; the report includes all TCA users if you include them in the report's user population.

**User Active Status**

Enter one of the following values to identify the user-account status.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Include active accounts, which belong to users with current roles.</td>
</tr>
<tr>
<td>I</td>
<td>Include inactive accounts, which belong to users with no current roles.</td>
</tr>
<tr>
<td>All</td>
<td>Include both active and inactive user accounts.</td>
</tr>
</tbody>
</table>

**Report Results**

The output is an XML-formatted file where user accounts are grouped by type, as follows:

- Group 1 (G_1) includes HCM user accounts.
- Group 2 (G_2) includes TCA party user accounts.
- Group 3 (G_3) includes OIM user accounts.
The information provided in the extract varies with the account type.

**HCM User Accounts**

**Business Unit Name**
The business unit from the primary work relationship.

**Composite Last Update Date**
The date when any one of a number of values, including assignment managers, location, job, and person type, was last updated.

**Department**
The department from the primary assignment.

**Worker Type**
The worker type from the user’s primary work relationship.

**Generation Qualifier**
The user’s name suffix (for example, Jr., Sr., or III).

**Hire Date**
The enterprise hire date.

**Role Name**
A list of roles currently provisioned to workers whose work relationships are all terminated. This value appears for active user accounts only.

**Title**
The job title from the user’s primary assignment.

**TCA User Accounts**

**Organizations**
A resource group.

**Roles**
A list of job, abstract, and data roles provisioned to the user.

**Managers**
The manager of a resource group.

**OIM User Accounts**

**Start Date**
The date from when the account existed.
Created By

The user name of the user who created the account.

FAQs for Manage Users

What happens if I send the user name and password?

An e-mail containing the user name and password is sent to the user’s primary work email address. If the user has no primary work email address, then the user name and password are sent to the primary work email address of the user’s line manager, if available; otherwise, no notification is sent.

You can select **Send user name and password** only if these details have not already been sent for this user: the user name and password can be sent once only for any user. If this option is available for selection but you do not select it, then you can run the process Send User Name and Password E-Mail Notifications later to notify users of their user names and passwords.

Can I extract details of all Oracle Fusion Applications users?

Yes. The Oracle BI Publisher User Details System Extract report includes details of all user accounts or a specified subset. For example, you can produce a report showing inactive user accounts, accounts created between specified dates, or accounts associated with TCA parties only.

To run the report, you must have an HCM data role that provides view-all access to person records for the Human Capital Management Application Administrator job role.
Common Applications Configuration: 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 predefine 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 Manger role entitles the user to enter orders, which could result in exploitation when filling shipping quotas. You can remove the entitlement to enter orders that the Order Manger job role inherits from the Orchestration Order Scheduling Duty role. Or you could segregate the shipping and order entry duties by defining an SOD policy that allows a user to have either job role but not both.

**False Positives**

False positives can be SOD policy violations that 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_PROGR</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.
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


  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: 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.

Set Help Options

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.

**Important**

For non-Cloud implementations only: Some help includes links to guides outside the help system. If you select this feature, then these links open guides on the Oracle Technology Network Web site. If you do not select this feature, then your system administrator must download the guides (http://download.oracle.com/docs/cds/E39540_01.zip) and put all the content from within the extracted E39540_01 folder directly into the appmgr/APPLTOP/fusionapps/applications/ahc/afh/reference/TechLib folder.

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.

**FAQs for Set Help Options**

**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.

**FAQs for Assign Help Text Administration Duty**

**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.

**Manage Help Security Groups**

**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_HUMANRESOURCE_SPECIALIST as the role name.
7. Click Save and Close.

You have created a new lookup code for the Help Security Groups lookup type, which is a standard lookup. The lookup code has the name, meaning, and description that you defined for the help security group.

You have also created a data security policy for the help database resource, specifying that the Human Resource Specialist role can view help that is defined with the HR security group. If you go to the Manage Database Resources and Policies page and find the database resource, or object, ATK_KR_TOPICS, then you can see the policy for the Human Resource Specialist role, with the condition that the column name, SECURITY_CODE, is equal to the value HR.

Help File Customization

Help File Customization: Overview

If you have the appropriate job roles, then you can customize the help files in the help site. Use the Manage Custom Help page to maintain both predefined and custom help files. You can create, duplicate, edit, and delete custom files, or
set their status to Active or Inactive. For predefined files, you can only duplicate them or set their status.

For each help file, predefined or custom, use help locations to determine where the help file appears in the application and in the help site. You have various options in how you add custom help, for example by uploading a file or specifying a URL. You can upload files of any type, and your URL can also point to files of any type.

**Note**

To make a copy of all custom help for testing, migration, or other purposes, create a configuration package then use the export and import feature in the Setup and Maintenance work area. The configuration package must use a source implementation project that contains the Define Help Configuration task list and you must select the following objects to export: Help Configuration and Help Topic.

**Customizing Help in Help Windows**

Many help files can be accessed from help windows in the application. If you want to customize help in the context of a help window, for example create a custom help file and add a link to it from a specific help window, then start by opening that help window. When you click the Manage Custom Help link, you go to the Manage Custom Help page, and the help location fields are automatically populated with values that correspond to the help window. This way you can easily select existing files to add to the same help location, and when you create a new file, the same help location appears by default.

**Restriction**

Aside from links to help files, some help windows also display informational text. You can't use the Manage Custom Help page to edit this text. Your technical administrators can do so using developer tools (not available in Oracle Cloud implementations).

**Customizing Help by Help Location**

Open the Manage Custom Help page directly from the home page of Oracle Fusion Applications Help or from search result pages.

**Tip**

When you search in the Manage Custom Help page, make sure that the Custom Help Only check box is not selected if you are looking for predefined help.

**Editing Specific Files**

To edit a specific file, you can either find it in the Manage Custom Help page, or open the file itself and click the Edit link.

**Customizing Glossary Terms**

You can find glossary terms in the Manage Custom Help page, or go to Navigator - Glossary in Oracle Fusion Applications Help to open the Glossary tab, search for the term, and click Edit.
Adding UPK to the Settings and Actions Menu

If your enterprise has purchased Oracle User Productivity Kit (UPK) content, then your administrator can also add a UPK item to the Settings and Actions menu in the global area of Oracle Fusion Applications. When users select this menu item, they access UPK content specific to the page that they are on.

Help Types: Explained

Oracle Fusion Applications Help contains various types of help content, including demos, examples, FAQs, glossary terms, help topics, and PDF guides. A business process or product can be supported by some or all of these help types.

Demo

Demos are Oracle User Productivity Kit (UPK) topics that visually demonstrate how to use the application to complete a short task or portion of a task. Demos can also provide an introduction to complex dashboards and work areas.

Example

Examples provide real use cases of features to illustrate how and when to use the feature, or scenarios to illustrate abstract concepts. Worked examples show exactly what you need to do to achieve a specific result, emphasizing decisions that you make and values that you enter.

FAQ

FAQs, or frequently asked questions, provide brief answers to questions that you might have regarding a task or page. For example, they can briefly explain what a term means, why something happened, how you can perform an action, or what happens if you perform the action.

Glossary

Glossary terms provide definitions for words or phrases used in help. You can search or browse glossary terms in the Glossary tab of Oracle Fusion Applications Help. Where the links are available, you can also see the definition when you hover over the term in help content for other help types.

Help Topic

Help topics explain key concepts, illustrate how application components work together, or assist in decision-making by explaining points to consider or the options you have. Help topics can also provide reference, overview, and other information.
PDF Guide

PDF guides in Oracle Fusion Applications Help provide, in a book format, information usually not found in other help types. There are other guides that present a collection of help content from the other help types, except demos, in an organized and logical format. These guides, for example, address specific business processes and setup offerings. You can find these guides by going to Navigator - Documentation Library in Oracle Fusion Applications Help.

Help Locations: Explained

Help locations determine where users can find help files, custom or not, from either the application or the help site.

Help locations include:

- Page or section values
- Help hierarchies
- Primary locations

Page or Section Values

The value in the Page or Section field on the help customization pages represents where users can click a help icon to open a help window that contains a link to the help file. In most cases, this value represents a page or region header in the application. Help windows are also available on specific tabs or windows, and in the Setup and Maintenance work area for specific task lists or tasks. You can associate a help file with multiple page or section values, or with none at all.

The page or section value reflects the logical navigation to the help window. For example, Edit Opportunity page, Revenue tab, Recommendations window does not mean that the help file is available in three different places. The help icon is in the Recommendations window, which is accessed from the Revenue tab on the Edit Opportunity page.

If the value suggests multiple locations, for example Create and Edit Opportunity pages, then the help file is available from the page header of both the Create Opportunity and Edit Opportunity pages. If the page or section value is, for example, a dashboard region that appears in multiple dashboards, then the value does not specify the page name but just the region. The help file is available from that region in multiple dashboards.

Help Hierarchies

Help files are associated with help hierarchies, which are used to categorize help files and aid users in finding help. Each help file can have multiple hierarchies, with at least one of type Business Processes. The business process hierarchy is based on the Business Process Management model. Every page or section value is predefined with a specific business process hierarchy. If you select a page or
section without entering a business process hierarchy, the predefined hierarchy
appears by default.

The Common Tasks navigator is based on the Welcome hierarchy type. The level
1 nodes represent categories of functional areas common to all users.

The Business Processes navigator in the help site is based on the business process
hierarchy. For example, if you assign two business process hierarchies to a help
file, users can find the file in both locations in the navigator. When the user clicks
More Help from a help window, all help files assigned to the same business
process hierarchy as the page or section value are returned as search results.

Similarly, the Products navigator is based on the Product hierarchy type, in
which level 1 is the product family, level 2 is the product, and level 3 is the
business activity owned by that product.

The Functional Setup navigator is based on the Functional Setup hierarchy type.
The level 1 nodes for this hierarchy are:

- Functional Setup Manager, which includes help about using the Setup
  and Maintenance work area.
- Offerings, which contains level 2 nodes for each setup offering, and lower
  levels for the main task lists in the offerings. Help for the task lists and
tasks are included.

Primary Locations

The primary location of a help file designates the hierarchy that is displayed
for the help file in search results and within the help content as breadcrumbs.
You cannot change the primary location of a help file that came with your help
installation. Primary locations of predefined help are based on the business
process hierarchy, while custom help files can have primary locations based on
hierarchies of any type.

Editing Predefined Help and Glossary Terms: Points to Consider

When you open any predefined help file, including glossary terms, that came
with Oracle Fusion Applications Help, you can see an edit option if you have
roles allowing edit access. When you edit predefined help, keep in mind:

- What happens to the original help file
- Where predefined help appears
- Considerations specific to glossary terms

What Happens to the Original Files

When you edit predefined help, you are actually creating a new custom help file
based on the original file, with the same help locations. The customized version
replaces the original, which becomes inactive and hidden from users. You can
display both versions by reactivating the original in the Manage Custom Help
page.

Note
In the Search Results: Existing Help region on the Manage Custom Help page, there is no option to edit predefined help. You can duplicate a predefined help file, edit the copy, and optionally inactivate the original.

**Where Predefined Help Appears**

All predefined help comes with preassigned help locations, including at least one based on the hierarchy of type Business Processes. Many also have predefined page or section values that indicate where the help can be accessed from help windows in the application.

To change where predefined help appears, either in the help site navigators or in the application, create a duplicate in the Manage Custom Help page. Change or add help locations to your custom copy, and inactivate the original.

Even though glossary terms do not appear in the help site navigators, you still need to enter at least one help location to categorize the glossary term.

**Considerations Specific to Glossary Terms**

When you edit a predefined glossary term, the original term becomes inactive. Existing links to the glossary term, from other predefined and custom help files, will automatically point to your custom version. If you later inactivate the custom glossary term, make sure to activate the original term so that the links still work.

**Links in Custom Help: Points to Consider**

When you create or edit custom help, follow best practices when you include links to help files or other content. If you are working on custom help created by duplicating a predefined help file, then you may see existing links from the original file in the Help Content section. The types of links that you can work with include:

- Related help links
- Standard hypertext links
- Links to documentation library content
- Glossary term links

For all link types, except the standard hypertext links, you must create or edit custom help with a Text or Desktop source type. In other words, you must type the help content directly in the application or use an HTML file that you upload to help. For standard hypertext links, the source type can also be URL.

**Related Help Links**

Related help is the section at the end of help files that contains links to other help files. The syntax for related help contains a comma-separated list of title IDs that represent help files.

This figure provides an example of related links code.
You can delete this code to remove all related help, or delete title IDs to remove individual links (for example, CREATE_AUTOMATIC_POSTING_CRITERIA_S_0000).

To replace existing links or add new links, you need to retain the code syntax and enter desired title IDs. To find title IDs, search for the help files on the Manage Custom Help page. Title IDs are displayed in the search results, but the Title ID column is hidden by default.

**Standard Hypertext Links**

You can create standard hypertext links to any file or Web site as long as you ensure the stability and validity of the links, including links to other help files, custom or not. These links can appear anywhere in the body of your help file as long as they come before any related help links.

In the Help Content section, highlight the text that you want to use as link text and click the **Add Link** icon button.

For links to other help files, open the file to which you want to link, and click the **E-Mail** link. Use the URL in the autogenerated e-mail text as the link to the file.

**Note**

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

**Links to Documentation Library Content**

The syntax for links to HTML files in documentation libraries is:

```html
<\?ofa linkToSTDoc(WCSUG4636) ?>Understanding Tags</\?ofa endLink >
```

WCSUG4636 is the anchor ID and Understanding Tags is the link text. You can:

- Modify the link by replacing the existing anchor ID or editing the link text, or both.
- Remove the link by deleting all the code for it.
- Create links to documentation library content by following the same syntax. These links can appear anywhere in the body of your help file as long as they come before any related help links.

**Important**

To ensure that you are linking to a supported documentation library, enter anchor IDs only from documentation libraries that are linked from predefined help topics.

**Glossary Term Links**

Glossary term links provide definitions in a note box when users hover over the term in help files.
This figure shows an example of code for a glossary term link.

```
<dfn><dfn>Glossary Term"accounting period", ACCOUNTING_PERIOD_001</dfn>
```

In this example, **accounting period** is the link text, or glossary term, and ACCOUNTING_PERIOD_001 is the identifier, or title ID.

- To remove the link but retain the text, delete all the code except the term itself.
- To add glossary term links, you must follow the link syntax and use the correct title ID for the glossary term. You can find title IDs in the search results of the Manage Custom Help page.

**Note**

If your custom help has glossary terms and the source type is Desktop File, then make sure before uploading that the quotes around the glossary term are actual quotation marks in raw HTML, not &quot; . Otherwise, quotation marks will appear when users view the help file.

---

**Customizing PDF Guides: Worked Example**

This example demonstrates how to customize a PDF guide that came with Oracle Fusion Applications Help. This guide is currently not available from any help window in the application.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What changes do you need to make to the guide?</td>
<td>Change the title of a chapter and remove a section in that chapter, to hide content about a particular subject</td>
</tr>
<tr>
<td>Which help window should the customized guide appear in?</td>
<td>The help window for the entire Welcome dashboard of Oracle Fusion Applications</td>
</tr>
<tr>
<td>Which help navigators should the customized guide appear in, and on which node?</td>
<td>Same as the original guide, plus the path associated with the help window</td>
</tr>
<tr>
<td>Do you want to limit access to the customized guide?</td>
<td>No, same as the original guide</td>
</tr>
</tbody>
</table>

Edit a copy of the original PDF guide, and use the Manage Custom Help page to replace the original PDF guide with your new file.

**Copying and Editing the PDF Guide**

1. Open the original PDF guide from the help site and save a copy to your desktop. Leave open the help file for the guide.
2. Using a PDF editor application, change the title of the chapter wherever the chapter title appears. Delete the content you want to hide from users.

**Replacing the Original PDF Guide**

1. In the help file that you still have open for the original PDF guide, click the **Edit** link.
2. On the Create Help page, use the default values except where indicated.
3. Update the title to the name that you want to display to users.
4. In the File Name field, browse for and select your customized guide.
5. Delete any keywords or parts of the description relevant to the content you removed from the PDF guide.
6. Add a help location with the Business Processes hierarchy type and select Information Technology Management as the level 1 node, Manage Enterprise Application Software as the level 2 node, and Use Applications as the level 3 node.
7. Select Welcome page in the Page or Section column.
8. Click Save and Close. The help file for the original PDF guide is automatically set to inactive.

**Adding Custom UPK Content to Help: Worked Example**

This example demonstrates how to add custom Oracle User Productivity Kit (UPK) topics as demo help files. These help files function like any predefined help file for demos. You can search and include these files in help windows and navigators as you would other help.

In this scenario, you are adding one demo about activity streams, to appear in a help window on the Welcome dashboard.

**Note**

Your demo must be made with UPK 3.6.1 or later to be added as help.

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What UPK content do you want to add to help?</td>
<td>From a UPK module containing five topics, add one as a custom demo on the help site</td>
</tr>
</tbody>
</table>
| Which help navigators should the demo appear in, and on which node? | Because the demo is about activity streams:  
  • Search by Common Tasks navigator, under the Collaboration node  
  • Search by Business Process navigator, under Information Technology Management - Manage Collaboration - Manage Collaborative Communities |
| Which help window should the demo appear in?  | On the Welcome dashboard of Oracle Fusion Applications, in the help window in the Activity Stream region |
| Do you want to limit access to the help file for the demo? | No |
| Do you want the help file to appear in the New and Updated pane? | Yes |
Generate a report of UPK document IDs, which you will use when creating custom help, to identify the UPK topic that you want to add. Publish the UPK module as a player package, then create custom help for the UPK topic that you want to use as a help demo.

**Generating a UPK Document ID Report**

1. In the UPK Developer, select **Details View**.
2. Right-click any column header, for example Name, and select **Column Chooser**.
3. In the Column Chooser dialog box, click and drag the Document ID column header and drop it after the Name column. Close the Column Chooser dialog box.
4. From the File menu, select to print, and save the output as a Microsoft Excel file to your desktop.

**Creating the Player Package**

1. From the UPK Developer, make sure that the topic that you want to add as a demo has the See It play mode. The topic can also have other modes, but only the See It mode is included in the custom help file.
2. Publish the module, specifying any location for the output and selecting to publish the selection only.
3. In the Formats section of the Publish Content window, select the **Player** check box under the **Deployment** check box group.
4. In the Player section, select the **Include HTML Web Site** check box, to ensure that the custom help file includes a text-only version of the UPK topic.
5. Finish the publishing process, after optionally setting other options.
6. Navigate to the location where you specified the output to be generated.
7. In the Publishing Content folder, copy the PlayerPackage folder and add it to the web server where you store UPK content.

**Creating Custom Help for Demos**

1. Open the help window in the Activity Stream region on the Welcome dashboard of Oracle Fusion Applications, and click **Manage Custom Help**.
2. On the Manage Custom Help page, the page or section and hierarchy values are populated with the values for the Activity Stream region.
3. Click **Create**.
4. On the Create Help page, complete the fields in the General Information section, as shown in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>The name of the UPK topic.</td>
</tr>
<tr>
<td>Source Type</td>
<td>Oracle User Productivity Kit</td>
</tr>
</tbody>
</table>
### File Location

The full URL of the player package folder on the Web server, for example, `http://<your domain>.com/UPKcontent/PlayerPackage`.

### Document ID

The document ID of the UPK topic to add to the help window in the Activity Stream region. You can copy and paste this ID from the Microsoft Excel file that you generated earlier.

### Help Type

Demo

### Help Security Group

Unsecured

### Keywords

Terms relevant to the demo.

### Description

Summary of the demo.

### Include in New and Updated pane

Selected

The Help Location section contains values for the help window in the Activity Stream region. This help file will also appear in the Search by Business Process navigator under this predefined hierarchy.

5. Click **Save and Close**.

6. On the Manage Custom Help page, open the help locations for the help file that you just created.

7. Add a help location with the Welcome hierarchy type and select **Collaboration Features** as the level 1 node.

8. Click **Save and Close**.

**FAQs for Help File Customization**

**How can I restrict help content to specific user roles?**

When you create or edit help, select a help security group that represents the set of roles that you want to have access to the help. If you do not see the Security Group field, then your administrator has not selected the Custom Help Security feature choice. The Unsecured group has no associated roles, so anyone can view the help. The predefined Secured group includes all internal employees and contingent workers, unless this group has been edited. You can create security groups and associate roles using the Manage Help Security Groups page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Help Security Groups task. Your new security groups are immediately available for use to secure new or edited help files.

**Why can't I select and add help to a location?**

You must specify a page or section to add the existing help to. To ensure that help is added to the correct help window, go to the page or section in the application, click the **Help** icon, and click the **Manage Custom Help** link in the help window. Alternatively, in the Manage Custom Help page, search for at least
a page or section and a level 1 value for the Business Processes hierarchy type before selecting the **Select and Add** option.

You cannot select and add help to a particular hierarchy, on the Manage Custom Help page, without a page or section. To add just a hierarchy, search for the help file, add a new help location, and specify only the hierarchy information.

**What happens to custom help when a help patch is applied?**

Oracle Fusion Applications Help patches update all help files, both active and inactive, except custom help. Custom help files are not affected by patches. Consider reviewing inactive files to see if you want to activate the updated version, or to make similar edits to the custom versions of those files, if any.

---

**Embedded Help Customization**

**Customizing Embedded Help: Highlights**

You can customize help that is embedded in the application, for example hints, for all users of Oracle Fusion Applications. There are different types of embedded help.

**Creating, Editing, or Deleting Help**

- Use Page Composer to edit, create, or delete hint text that appears on hover over buttons, links, icons, or tab titles. Open the properties of the UI element to define the help text in the shortDesc field. Usually, the value resolves to a key in a resource bundle.

- Edit, create, or delete other types of embedded help. Refer to the Customizing or Adding Static Instructions, In-Field Notes, and Terminology Definitions section.

  See: Oracle Fusion Applications Extensibility Guide for Developers
Common Applications Configuration: 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 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.
Common Applications Configuration: 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 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
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

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

**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 business objects are grouped by modules. The available enablement options are:
   - **Manual**: (Recommended) Empowers the user to decide whether to share each instance of the object with social network.
   - **Automatic**: Automatically sends the newly enabled object instances and updates 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.

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.

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.
Oracle Sales Cloud Computer Telephony Integration: Explained

You can use Oracle Sales Cloud’ Computer Telephony Integration (CTI) to place a call to a contact from a hyperlink on the phone number or phone icon.

Here are a few topics that are important to know when using CTI:

- Normal call flow
- Interaction Records and Notes
- Operational Notes

Note

CTI must be enabled to make calls using the various contact information pages and pop-up UIs. When enabled, phone numbers appear as hyperlinks. Interaction logging is available if that feature is enabled. If interaction logging is available, a note indicating that fact will be displayed.

Normal Call Flow

CTI uses a call-the-caller-then-call-the-callee procedure for completing a phone call. That format and the normal flow of this procedure are described below.

- You initiate a call
  
  If you see a small orange square next to a contact or customer name, click the square to display further details, including phone numbers. To place a call, place your mouse over the phone number hyperlink and click.

Note

CTI does not work on phone numbers that are marked with a Do Not Call icon.

- Select a Calling Phone
Choose the calling phone number. Usually the calling phone is a number from your profile information. Alternately, if you need to use a phone not in your profile, you can specify a different number to originate your call.

- **Call Flow**

After you select the calling phone number, the system calls you back on that number, waits for you to answer, and then calls the person for whom the call is intended.

**Interaction Records and Notes**

CTI automatically creates an interaction record of the call, when that feature is enabled. The details window that provides the phone number may also show an Interaction icon that you can click to display a list of interaction records to edit, for example to provide a description of the call. The window may also provide a notes feature that you can use to record notes during the call.

**Interaction Logging**

The interaction record is logged as soon as the call is either successfully set up or known to have failed.

The interaction log records the customer, call participants, a timestamp noting the start time of the call, the direction of the communication, in or outbound, and
the resolution code. The description is automatically updated with these three items:

- Call ID from OWLCS
- Your chosen phone number
- Contact phone number

The call resolution code is determined from OWLCS and recorded in the interaction:

<table>
<thead>
<tr>
<th>OWLCS Call Status</th>
<th>Resolution Code in Interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallConnected</td>
<td>CALL ANSWERED</td>
</tr>
<tr>
<td>CallAborted</td>
<td>FAILED</td>
</tr>
<tr>
<td>CallHangUp</td>
<td>FAILED</td>
</tr>
<tr>
<td>CalledPartyBusy</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>CalledPartyNoAnswer</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>CalledPartyNotReachable</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>CallingPartyBusy</td>
<td>FAILED</td>
</tr>
<tr>
<td>CallingPartyNoAnswer</td>
<td>FAILED</td>
</tr>
<tr>
<td>CallingPartyNotReachable</td>
<td>FAILED</td>
</tr>
</tbody>
</table>

**Editing interactions**

Once the call is established, if Interactions is available, you can use the Interactions icon on the UI to launch the interaction record list view. Select the current interaction record to edit it.

**Operational Notes**

Because of the call-the-caller-then-call-the-callee format, there are some conditions that may occur due to several calling situations. Some of these conditions are described below:

- Why don’t I hear a ring-back tone? As soon as you answer the system call-back, the system immediately dials the contact. You won’t hear a ring-back tone as in a normal outbound phone call. However, you can tell that the call attempt is progressing because:
  - The phone indicates that the connection is active. If the call to the contact reaches a busy tone or the call attempt times out, the connection is dropped.
  - The dialing window stays on the screen while the call attempt is progressing. It disappears when the connection is either successfully established or fails.

- What if your phone is busy and the call-back goes directly to voice mail? Normally this would not happen because you would not initiate a new call when you are already busy on another call. However, this situation could occur due to a race condition, that is where another incoming call reaches your phone before the CTI call-back. When this happens, two different scenarios could occur:
• If your phone is configured for busy-forward-all-to-voice-mail, the CTI call would be forwarded to your voice mail, and the system thinks that the caller has answered the call and will proceed to call the contact. On answering, the contact hears your voice-mail greeting.

• If your phone is capable of presenting a second call to the user, as is supported by many office phones and mobile phones, then you can still answer the CTI call and there is no issue.

• What if you wait too long to answer the call-back? In other words, you wait longer than the ring-no-answer-forward-to-voice-mail timer on the phone system and the call goes to voice mail. Normally, this would not happen because you are expecting the inbound call after you started the call, and would answer promptly. However, if for some reason you do not answer and allow the call to ring-no-answer-forward to voice mail, then the system would think that you have answered the call and will proceed to call the contact. On answering, the contact hears your voice-mail greeting.

• What if the contact does not answer in 30 seconds and the system abandons the call attempt? If the contact’s voice mail is configured to answer after 30 seconds, you will not be able to leave a message.

Oracle Sales Cloud CTI: Top Tasks

Oracle Sales Cloud Computer Telephony Integration (CTI) is a feature of the customer contact process. Phone communication to customers and employees is initiated with a click of the mouse, leveraging your customer contact information and the application context. The CTI feature uses Oracle WebLogic Communication Services, OWLCS, to enable communications. Applications that provide the CTI functionality do so primarily through contextual actions.

Additionally, CTI utilizes Oracle Sales Cloud interactions as an optional transaction logging feature that will track information about the call such as the customer, call participants, a timestamp noting the start time of the call, the direction of the communication, in or outbound, and the resolution code.

CTI integrates with your telephony environment and must be manually enabled in your deployment. This topic highlights what is required to set up the CTI feature and to implement logging of the calls made using the CTI feature.

Terms used in setting up these communications

• PSTN: Public switched telephone network is the network of the world’s public circuit-switched telephone networks.

• SIP: Session initiation protocol, an open signaling protocol standard that is used to set up phone calls

• TPCC: Third Party Call Control enables an application to control the telephony network to set up calls automatically.

• OWLCS: Oracle WebLogic Communication Services. Offers the TPCC service to Oracle applications and sets up the calls via SIP integration with the telephony network.
The set up task list Define WebLogic Communication Services Configuration delineates four tasks required for the correct configuration and implementation of CTI. There is an optional task, separate from the set up task list, required for implementing Interaction logging.

Information about implementing CTI can be found in the Oracle Sales Cloud Administrator's Guide. Detailed information about configuring and maintaining WebLogic Communication Services is found in the Oracle WebLogic Communication Services Administrator's Guide

**Configure and Deploy WebLogic Server**

- Deploy WebLogic Communication Services: After the Oracle WebLogic communication server is deployed, this manual task activates the server.

  See: Oracle WebLogic Communication Services Administrator's Guide

**Integrate Communications Services**

- Integrate WebLogic Communication Services with Telephony Network: This manual task integrates communications within the telephony environment. OWLCS must be configured to interface with the specific characteristics of the telephony network.

  See: Managing Oracle WebLogic Communication Services for CTI Functionality

**Specify the Domain and Address**

- Register a URL for the telephony gateway or soft switch for SIP domain: This task defines the Server protocol, defaulted to http, the external server host address and external server port address. The Associated Modules section is not required for setup. You can also perform this as a manual task using Topology Manager to configure the address of the SIP Public Switched Telephone Network (PSTN) gateway or SIP soft switch serving the users within that domain. This address is needed by CTI to correctly form the SIP addresses required by WebLogic Communication Services.

  See the link to Configuring PSTN Gateway Address Using Topology Manager: Worked Example.

**Enable Click-to-Dial**

- After configuring the server and defining the SIP domain, perform the Enable Click-to-Dial task. This task sets the value of the profile option Enable Click-to-Dial to 'Yes.'

**Call Logging via Interactions**

- To initiate the Interaction based logging for CTI, set the profile option Call Interaction Logging Enabled to 'YES.'

**Configuring PSTN Gateway Address Using Topology Manager: Worked Example**

This example demonstrates how, during the implementation of the Register URL for the telephony gateway or soft switch for SIP domain task,
you must manually configure the PSTN gateway address by registering HzCTDPstnGatewayApp to a given environment using Oracle Fusion Topology Registration.

These steps configure the address of the SIP Public Switched Telephone Network (PSTN) gateway or SIP soft switch serving the users within that domain. This address is needed by Click-to-Dial to correctly form the SIP addresses required by WebLogic Communication Services.

For example: SIP:+1650-555-1212@pstn_gateway.oracle.com;user=phone where pstn_gateway.oracle.com is the SIP domain. The SIP domain can also be expressed in the format 10.1.1.1 (IP address).

**Configuring PSTN using the Topology Manager**

1. Sign in to Oracle Fusion Applications as a user that has **application implementation consultant** and **WebLogic Services administration** roles.

2. In Fusion Applications Setup and Maintenance, click Register Enterprise Applications from the regional area under **Topology Registration**.

3. On the Register Enterprise Applications page, click the plus icon to add an enterprise application. An Add Enterprise Application popup appears.

4. Enter the new application information: Click Search in the Enterprise Application list field. Enter HzCTDPstnGatewayApp in the name field and click Search. Click OK.

5. Enter the other fields in the Add Enterprise Application popup.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>HzCTDPstnGatewayApp</td>
</tr>
<tr>
<td>Server Protocol</td>
<td>SIP</td>
</tr>
<tr>
<td></td>
<td>This field is ignored by click-to-dial. Oracle WebLogic Communication Service (OWLCS) always uses the SIP protocol.</td>
</tr>
<tr>
<td>External Server Host</td>
<td>10.143.167.91 (Used as an example)</td>
</tr>
<tr>
<td></td>
<td>A host name can be used instead of an IP address.</td>
</tr>
<tr>
<td>External Server Port</td>
<td>0 (Used as an example)</td>
</tr>
<tr>
<td></td>
<td>This field is ignored by Click-to-Dial.</td>
</tr>
</tbody>
</table>

6. Click Save and Close.
Define 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.
## Define Applications Core Configuration

<table>
<thead>
<tr>
<th>Lookup</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>Lists the available codes and translated meanings</td>
</tr>
<tr>
<td>Set enabled</td>
<td>Additionally associates a reference data set with the lookup codes</td>
</tr>
<tr>
<td>Common</td>
<td>Legacy lookups</td>
</tr>
</tbody>
</table>

Standard lookups are the simplest form of lookup types consisting only of codes and their translated meaning. They differ from common lookups only in being defined in the standard lookup view.

Common lookups exist for reasons of backward compatibility and differ from standard lookups only in being defined in the common lookup view. Set enabled lookup types store lookup codes that are enabled for reference data sharing. At runtime, a set-enabled lookup code is visible because the value of the determinant identifies a reference data set in which the lookup code is present.

### Accessing Lookups

Standard, set-enabled, and common lookups are defined in the Standard, Set-enabled, and Common views, respectively. Applications development may define lookups in an application view to restrict the UI pages where they may appear.

In lookups management tasks, lookups may be associated with a module in the application taxonomy to provide a criteria for narrowing a search or limiting the number of lookups accessed by a product specific task such as Manage Purchasing Lookups.

### Enabling Lookups

A lookup type is reusable for attributes stored in multiple tables. Enable lookups based on the following.

- Selecting an Enabled check box
- Specifying an enabled start date, end date, or both
- Specifying a reference data set determinant

If you make changes to a lookup, users must sign out and back in before the changes take effect. When defining a list of values for display rather than validation, limit the number of enabled lookup codes to a usable length.

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.
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 view. Lookups defined in an application view can only be managed by following instructions provided by the owning application.

**Note**
A lookup type and its codes can only be defined in one lookup view.

---

**What's the difference between a lookup type and a value set?**

A lookup type consists of lookup codes that are the values in a static list of values. Lookup code validation is a one to one match.

A table-validated value set can consist of values that are validated through a SQL statement, which allows the list of values to be dynamic.

**Tip**
A table validated value set can be defined based on any table, including the lookups table. This allows a lookup type to be made into a table-validated value set that can be used in flexfields.

<table>
<thead>
<tr>
<th>Area of Difference</th>
<th>Lookup Type</th>
<th>Value Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of values</td>
<td>Static</td>
<td>Dynamic if Table validation type</td>
</tr>
<tr>
<td>Validation of values</td>
<td>One to one match of meaning to code included in a lookup view, or through the determinant of a reference data set</td>
<td>By format or inclusion in a table</td>
</tr>
<tr>
<td>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_CMN, 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.
• Open: Indicates that the tree is associated with all reference data sets.
• Set ID: Indicates that the tree will be associated with a specific reference data set.

Creation Mode

Indicates the source where the tree structure is being defined. For predefined tree structures select Oracle and for custom structures, select Customers.

Customization

You can customize the predefined tree structures as well as the ones that you created. However, customizing the predefined tree structures involves certain level of access restrictions, and will be limited to specific tree nodes and downwards in hierarchy.

Multiple Tree Versions

One or more trees and tree versions can be based on a tree structure. A tree structure can have one or more trees and tree versions based on it. Usually, only one active version is permitted at any given point of time. However, depending on the requirement, you can allow two or more tree versions to be in the active state for the same date range. This flexibility allows you to choose the tree version that you want to implement.

Managing Tree Structures: Points to Consider

You can create, edit, and delete tree structures depending upon the requirement. You can also audit and change the status a tree structure.

Creating and Editing Tree Structures

You can create trees on the basis of a tree structure. When you edit an active tree structure, the status of the tree structure and all associated trees and their versions change to draft. To reuse a tree structure, you can create a copy of it without copying the associated trees and tree versions. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.

Note

For specific information on working with the predefined tree structures that exist in an Oracle Fusion application, refer to the specific product documentation.

Setting Status

If you change the status of a tree structure, the status of the trees and tree versions associated with that tree structure also changes.
The following table lists the different statuses of a tree structure.

<table>
<thead>
<tr>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>Yet to be published or is in a modified state.</td>
</tr>
<tr>
<td>Active</td>
<td>In use and based on which one or more trees or tree versions are created.</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------</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>Restrict By Set ID</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Set ID check box is selected to enable the <strong>Restrict Tree Node List of Values Based on</strong> 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>Row Flattened Table Name</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.</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.** |
| **Restrict by Date** | **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.** | 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. | **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.** |
| **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.appcore.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>Column 1</td>
<td>Column 2</td>
<td>Column 3</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td><strong>Root Node</strong></td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the <strong>Allow Multiple Root Nodes</strong> check box for the <strong>Restrict Tree Node List of Values Based on</strong> 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><strong>Data Source Max Depth</strong></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 <strong>Maximum Depth</strong> field is set to <strong>Unlimited</strong>.</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><strong>Duplicate Node</strong></td>
<td>On the Data Source dialog box, if the <strong>Allow Duplicates</strong> 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 Set ID 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 <strong>Date Range</strong> check box is selected to enable the <strong>Restrict Tree Node List of Values Based on</strong> 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 <strong>Allow Multiple Active Tree Versions</strong> 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 <strong>Allow Range Children</strong> 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>
### Trees and Data Sources: How They Work Together

Data sources form the foundation for tree management in Oracle Fusion Applications. Tree structures, trees, and tree versions establish direct and real-time connectivity with the data sources. Changes to the data sources immediately reflect on the Manage Trees and Tree Versions page and wherever the trees are being used.

**Metadata**

Tree structures contain the metadata of the actual data that is used in Oracle Fusion Applications. Tree structures contain the core business logic that is manifested in trees and tree versions.

**Data Storage**

Trees and tree versions are built upon the tree structures. They employ the business rules defined in the tree structures and allow an application to select and enable a subset of trees to fulfill a specific purpose in that application.

**Access Control**

Source data is mapped to tree nodes at different levels in the database. Therefore, changes you make to the tree nodes affect the source data. Access control set on trees prevents unwanted data modifications in the database. Access control can be applied to the tree nodes or anywhere in the tree hierarchy.

### Adding Tree Nodes: Points to Consider

Tree nodes are points of data convergence that serve as the building blocks of a tree structure. Technically, the node may be stored either in a product-specific
table or in an entity that has been established by tree management as the default storage mechanism. However, since all data in Oracle Fusion Applications usually have a storage home, only user-created data needs to be stored in an entity. Nodes are attached to tree versions. Whenever you create or edit a tree version, you need to specify its tree node.

Managing Tree Nodes

You can create, modify, or delete tree nodes on the Tree Version: Specify Nodes page. To add a tree node, ensure that the tree structure with which the tree version is associated is mapped to a valid data source. You can also duplicate a tree node if the multiple root node feature is enabled.

Node Levels

In most trees, all nodes at the same level represent the same kind of information. For example, in a tree that reflects the organizational hierarchy, all division nodes appear on one level and all department nodes on another. Similarly, in a tree that organizes a user's product catalog, the nodes representing individual products might appear on one level and the nodes representing product lines on the next higher level. When levels are not used, the nodes in the tree have no real hierarchy or reporting structure but do form a logical summarization structure. Strictly enforced levels mean that the named levels describe each node's position in the tree. This is natural for most hierarchies. Loosely enforced levels mean that the nodes at the same visual level of indentation do not all represent the same kind of information, or nodes representing the same kind of information appear at multiple levels. With loosely enforced levels, users assign a level to each node individually. The level is not tied to a particular visual position.

Node Types

A tree node has the following node types.

- Single: Indicates that the node is a value by itself.
- Range: Indicates that the node represents a range of values and possibly could have many children. For example, a tree node representing account numbers 10000 to 99999.
- Referenced Tree: Indicates that the tree node is actually another version for the tree based on the same tree structure, which is not physically stored in the same tree. For example, a geographic hierarchy for the United States can be referenced in a World geographic hierarchy.

Define Profile Options

Profile Options: Explained

Profile options manage configuration data centrally and influence the behavior of applications. Profile options serve as permanent user preferences and
application configuration parameters. You configure profile options with settings for specific contexts or groups of users. Users customize how their user interfaces look and behave by changing the values of available profile options.

Profile options store the following kinds of information.

<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 updateable.

**Profile Option Definition and Configuration**

Application developers add new profile options and configure ones that are not to be updated by other users. Application administrators and implementation consultants configure profile options with profile option values that are implemented as updatable.

Profile option definitions consist of the following.

- Profile option name
- Application and module in the application taxonomy
- Profile option values
- Profile options categories
- Profile option levels
- Profile option level hierarchy

Profile options can appear on any user interface page without indication that a profile option is what is being set.

**Profile Option Values**

Some profile options have predefined profile option values.

The Manage Profile Option Values task flow allows an administrator to set updatable profile option values at the available levels, including the user level. You can access the Manage Profile Option Values task starting in the Setup and Maintenance Overview page and searching for profile option tasks.

You can set profile option values at different levels: site, product, and user. The following table provides examples.
Context such as user session or accessed product determines which profile option value is associated with the profile option name. In the example, if manager1 does not set a profile option value for this profile option, access to Financials for EMEA shows currency in Euros; and access to other products shows currency in UK pounds sterling.

**Profile Option Categories**

Categories group profile options based on their functional area. Profile option categories facilitate searching and defining data security.

For example, in Oracle Fusion Receivables, the Transactions profile option category groups profile options related to setting how Receivables transactions are to be processed, such as Require Adjustment Reason.

A profile option can be in more than one category.

**Profile Option Hierarchies and Levels**

Application developers specify at which hierarchy level a profile option is enabled. The predefined profile option hierarchy levels are site, product, and user.

The hierarchy levels specified in the profile option definition determine the context in which a profile option value may be set. If the profile option value at a particular level is updatable, an administrator can update the profile option value for that context.

**Note**

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

For security, one level in the hierarchy is designated as a user level. A profile option may be enabled at any or all hierarchy levels. When enabled at all levels,
the predefined ordering of profile option hierarchy levels gives precedence to the values that are set at the user level over values set at the product and site levels, and precedence to values set at the product level to values set at the site level. If there is no value for the current user, then the product value applies. If there is no value for the user or product, then the site value applies.

The table shows the predefined profile option hierarchy and ordering.

<table>
<thead>
<tr>
<th>Hierarchy Level</th>
<th>Priority When Multiple Levels Set</th>
<th>Effect on Applications</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>Lowest</td>
<td>Affect all applications for a given implementation</td>
<td>Currency for the site is set to Euros.</td>
</tr>
<tr>
<td>Product</td>
<td>Supersedes Site</td>
<td>Affect all applications of a product family such as Financials</td>
<td>Currency for the Financials products set to UK pound sterling.</td>
</tr>
<tr>
<td>User</td>
<td>Highest, supersedes Product</td>
<td>Affect only the experience of the current user</td>
<td>Currency for the user of Financials applications set to US dollars.</td>
</tr>
</tbody>
</table>

You can configure updatable values for profile options at one or more levels depending on which levels are enabled in the profile option definition. When a profile is set at more than one level, higher levels of specificity override lower levels of specificity.

In the example, if the currency setting for the site is UK pounds sterling, but the Financials division works in the Netherlands using the Euro, a manager in the US can override that product level setting at the user level to use US dollars when accessing Financials applications.

In another example, if a profile option called Printer is set only at the site and product levels. When a user logs on, the Printer profile option assumes the value set at the product level, since it is the highest level setting for the profile.

Tip

Set site-level profile option values before specifying values at any other level. The profile option values specified at the site-level work as defaults until profile option values are specified at the other levels.

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
Define Applications Core Configuration

• 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>
<tr>
<td>Product</td>
<td>CRM Application Composer</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Application Administrator</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Hima</td>
<td>Hindi</td>
</tr>
</tbody>
</table>

Values at the site level take 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 changed. Profile option changes do not affect C or PL/SQL processes that are already running.

Define Flexfields

Flexfields: Overview

A flexfield is an extensible 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
- Run time 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 context-sensitive so they appear only when needed on a row-by-row basis when specific contextual information is met. Extensible flexfields are similar to descriptive flexfields, but provide additional advanced features. Generally, setup
of descriptive and extensible flexfields is optional because their segments capture 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 run time 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.

**Run time 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 run time 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.

**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.
Oracle Procurement Cloud Implementing Procurement

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
- Run time
- 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 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 run time 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.

**Run time**

For a flexfield to reflect the latest flexfield definition at run time 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.

**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 run time 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 run time 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 run time.

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 run time users.

In the Define Flexfields tasks, you can deploy a flexfield using either of the following commands:
Define Applications Core Configuration

- 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 Settings and Actions - Personalization - Set Preferences in the global area and changing the text to the translated text for that locale.

Identifying Flexfields on a Run time 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 run time 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 run time 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.
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></td>
<td>If description help text is specified, a Help icon button is displayed next to the field in the run time application. The description help text is displayed when the user hovers over the Help icon button.</td>
</tr>
</tbody>
</table>
The field-level instruction help text to display for the field.

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.

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 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.
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 run time 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 run time behavior</th>
<th>Run time 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>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.

**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 run time 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 run time.

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 run time 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 run time environment.</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</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 run time 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.
Define Applications Core Configuration

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 run time 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 run time 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 run time.
Define Applications Core Configuration

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 run time 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:
```
sh $JDEV_HOME/oracle_common/common/bin/wlst.sh
```

Windows:
```
wst.cmd
```

Connect to the server, replacing the user name and password arguments with your WebLogic Server user name and password.
```
connect('wls_username', 'wls_password', 'wls_uri')
```

The values must be wrapped in single-quotes. The `wls_uri` value is typically `T3://localhost:7101`.

For more information on the WLST scripting tool, see the Oracle Fusion Middleware Oracle WebLogic Scripting Tool.

Using the `deployFlexForApp` Command

The `deployFlexForApp` command translates the product application’s predefined flexfield metadata into artifacts in the MDS repository.

**Important**

This command is run automatically when you provision applications. However, after custom applications development, you must run the `deployFlexForApp` command.
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. The values must be wrapped in single-quotes.

deployFlex('flex_code', 'flex_type')

Optionally, execute the following command if the flexfield is an extensible flexfield, and you want to deploy all the flexfield’s configurations.

**Note**

By default, extensible flexfields are partially deployed. That is, only the pages, contexts, or categories that had recent changes, are deployed.

deployFlex('flex_code', 'flex_type', ['force_Complete_EFF_Deployment'])

where, forceCompleteEFFDeployment=None
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 flexfields 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.
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 run time 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 many 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
Define Applications Core Configuration

13- 67

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 run time to display the corresponding translated text from the value column for the run time 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.

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>Run Time 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.

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 run time 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.
  - \{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.

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 run time 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 run time 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 criterion. 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 its 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.

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 run time 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>
<tr>
<td>Each usage or reference to the key flexfield</td>
<td>Application development</td>
</tr>
<tr>
<td>Structure instance</td>
<td>Administrators and implementation consultants</td>
</tr>
<tr>
<td>Other</td>
<td>Administrators and implementation consultants</td>
</tr>
</tbody>
</table>

If your key flexfield or certain usages or references of the key flexfield 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.

**Managing Key Flexfields: Points to Consider**

Consider the plans for a key flexfield, security, and resulting run time pages when configuring key flexfields.
Planning

Plan structures carefully and allow for future needs.

Caution

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.

Run time 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 run time.

**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 run time 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 run time 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.

**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
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 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>_<code>` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>_c` attribute contains 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.
FAQs for Define Flexfields

Why did my flexfield changes not appear in the run time 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 run time 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 run time correspond to the flexfield definition that was last deployed successfully.

What happens if a value set is security enabled?

Value set security is a feature that enables you to secure access to value set values based on the end user’s role in the system.

As an example, suppose you have a value set of US state names. When this value set is used to validate a flexfield segment, and users can select a value for the segment, you can use value set security to restrict them to selecting only a certain state or subset of states based on their assigned roles in the system.

For example, Western-region employees may choose only California, Nevada, Oregon, and so on as valid values. They cannot select non-Western-region states. Eastern-region employees may choose only New York, New Jersey, Virginia, and so on as valid values, but cannot select non-Eastern-region states. Value set security is implemented using Oracle Fusion Applications data security.

How can I set a default value for a flexfield segment?

When you define or edit a flexfield segment, you specify a default value from the values provided by the value set assigned to that segment.

You can set the default value 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 run time are reflected in the value of the segment.

If the display type of the segment is a check box, you can set whether the default value of the segment is checked or unchecked.
Define Attachments

Attachments: Explained

Attachments are pieces of supplementary information that users can associate with specific business objects such as expense reports or purchase orders. Attachments can be URLs, desktop files, text, or in cases where available, repository folders. For any given business object, a user may be able to only view attachments, or also create, delete, or edit attachments, depending on security. For more information on an introduction to attachments, see the Oracle Fusion Applications Developer’s Guide.

Repository

Attachments are stored in a content 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.
Common Setup: Other Common Setup and Maintenance Tasks

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.

Access to Person Data

Assigning Security Profiles to Job Roles for Oracle Fusion Transactional Business Intelligence Users: Explained

Users of Oracle Fusion Transactional Business Intelligence (Transactional Business Intelligence) need access to some person data for reporting purposes. To provide this access, you assign a predefined security profile to relevant job or abstract roles using the Oracle Fusion Human Capital Management (HCM) setup task Manage Data Role and Security Profiles. On completion of this task, Oracle Fusion Data Security is updated automatically for roles being used to access Transactional Business Intelligence.

Job or Abstract Roles and Related Security Profiles

The following table identifies, by Oracle Fusion product, the job and abstract roles that need access to person data and the predefined security profile that you assign to each role.

<table>
<thead>
<tr>
<th>Product</th>
<th>Job or Abstract Role</th>
<th>Security Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Fusion Assets</td>
<td>Asset Accountant</td>
<td>View All Workers</td>
</tr>
</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.
Decisions to Consider | In This Example
---|---
What is the name of the Transactional Business Intelligence job or abstract role? | Warehouse Manager
What is the name of the person security profile? | View All Workers

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: 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.

See: Customizing Existing Oracle Enterprise Scheduler Job Properties

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

Customization and Sandboxes

Customizing Simplified Pages Using Page Composer: Points to Consider

On a simplified page, you can click your name and select **Customize User Interface** to customize the UI using Page Composer. When customizing a simplified page, consider the customization layer to choose, the types of customizations you can make, and labels for your saved changes.

Customization Layers

The customization layer that you select before making changes to the page determines the scope of users impacted by your customizations. If you are not presented with customization layers to choose from after you select **Customize User Interface**, then your changes are made to the site layer. For more information to understand customization layers, see the Oracle Fusion Applications Extensibility Guide for Business Analysts.

Types of Customizations

In simplified pages, customization using Page Composer is limited to what you can change with component properties. For example, you can show or hide fields or make a check box required, but you cannot add new components or change the layout of the page.

After you select a customization layer, if available, you can click:

- **Design** to navigate around and get to the components you want to customize. You cannot make any customizations in this mode.
- Select to select a component on the page and open its properties.

Each component has its own set of properties, which may include some of the properties in this table.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Text used by screen readers, for information in addition to what is provided in the Short Desc property.</td>
</tr>
<tr>
<td>Label</td>
<td>Display text for the component, for example the field prompt or the single prompt for a group of check boxes</td>
</tr>
<tr>
<td>Read only</td>
<td>Whether users can edit the component, for example if a check box can be selected or not</td>
</tr>
<tr>
<td>Rendered</td>
<td>Whether the component is visible or hidden to users on the page</td>
</tr>
<tr>
<td>Required</td>
<td>Whether users must enter something for the component before saving the page</td>
</tr>
<tr>
<td>Short Desc</td>
<td>Text that appears when users hover or focus on the component, for example hover over a field label or click in the text box</td>
</tr>
<tr>
<td>Show Required</td>
<td>Whether an asterisk is displayed to indicate that the component is required</td>
</tr>
</tbody>
</table>

When you access component properties on a workstation page using Page Composer, more properties are available.

**Save and Label**

When you save, you can label your changes so that you can later revert to your saved customizations. Labels are stored with a prefix of `composer_`. For example, if you enter `myLabel`, then the label is `composer_myLabel`.

As needed later, you can click your name in the global area and select Manage Customizations. Click Promote for the desired component and select the label to revert to.

**Sandbox: Highlights**

Use a sandbox to commit customizations to a runtime use session for validation before deploying changes to the mainline. Administrators create and manage sandboxes. An active sandbox isolates changes from the mainline and other users.

Sandboxes can contain the following types of customization changes.

- Metadata, such as non-flexfield UI page customizations
- Data security
- Generated flexfields business components
Metadata changes are captured in a metadata sandbox. Data security changes are additionally captured in a data security enabled sandbox. Changes to a flexfield are captured in a flexfield that is deployed as a single flexfield sandbox. Once you are ready to make sandbox changes available in the mainline, you either publish the metadata or data security sandbox, or deploy the flexfield. Only metadata and data security sandboxes can be downloaded as a sandbox file for import to another Oracle Fusion Applications instance.

The following table lists the differences among the types of sandboxes.

<table>
<thead>
<tr>
<th>Type of Changes</th>
<th>Type of Sandbox</th>
<th>Method for Making Changes Available in Mainline</th>
<th>Downloadable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>Sandbox</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Data security</td>
<td>Sandbox enabled for data security changes</td>
<td>Publish sandbox</td>
<td>Yes</td>
</tr>
<tr>
<td>Flexfield</td>
<td>Flexfield deployed as a flexfield-enabled sandbox</td>
<td>Deploy flexfield</td>
<td>No</td>
</tr>
</tbody>
</table>

Only one sandbox can be active at a time. Changes made while a sandbox is active are captured in that sandbox.

For more information on using the Sandbox Manager, and customizing and securing pages, business objects, data, and custom objects in a sandbox, see the Oracle Fusion Applications Extensibility Guide.

**Managing a Page Customization Sandbox**

You can make metadata (MDS) type changes in a sandbox, including menu customizations, changes to the personalization menu, implicit ADF customizations, or changes made with Oracle Composer or CRM Application Composer.

- If you are entitled to do so, manage sandboxes in the Sandbox Manager.
  
  See: Using the Sandbox Manager

- Implement customizations on an existing page to change the metadata of a sandbox before deploying the changes to the mainline.

  See: Customizing Existing Pages

- Using CRM Application Composer, customize business objects in a sandbox before deploying the changes to the mainline.

  See: Customizing Objects

**Managing a Data Security Sandbox**

You can create a sandbox for data security testing, or designate an existing sandbox to become enabled for data security testing.

- If you are entitled to do so, manage data security-enabled sandboxes in the Sandbox Manager.
See: Using the Sandbox Manager

- If you customize business objects in CRM Application Composer, you may need to define data security policies to make them accessible to users.
  
  See: Defining Security Policies for Business Objects

- If you create new business objects, you need to secure them.
  
  See: Customizing Security for Custom Business Objects

**Managing a Flexfield Sandbox**

You create a flexfield-enabled sandbox by deploying one flexfield to a sandbox using the Manage Flexfield task flow. The flexfield sandbox gets its name from the flexfield you deploy. You cannot test two flexfields in the same sandbox. Once you deploy a flexfield as a sandbox, you must sign out and back in to view how the sandbox runtime reflects the flexfield changes, such as new segments. You can redeploy the same flexfield to the same sandbox repeatedly as you make incremental changes to the flexfield setup.

- Since a flexfield sandbox cannot be published, any page customizations or data security in the flexfield sandbox cannot reach the mainline when the flexfield is deployed to the mainline. If you have entitlement to do so, see Deploying a Flexfield to a Sandbox: Points to Consider.

- If you are entitled to do so, manage flexfield-enabled sandboxes in the Sandbox Manager.

  See: Using the Sandbox Manager
Define Basic Catalogs

Create Catalogs

Catalogs: How They Work Together

A catalog is a collection of categories that you use to classify items. You can organize the categories into a hierarchy that represents a taxonomy. You create new categories only in the context of a catalog. You can add existing categories to one or more catalogs, either from another catalog or as shared categories from a source catalog.

You can set the Catalog Content value to Items at all levels which allows items to be assigned to any level within the category hierarchy, not only to the leaf levels.

The following diagram shows the relationships of the catalog components.
Catalog
A catalog is a collection of categories that are organized to define a classification of items. The top most level of a catalog is the catalog root. All categories for the first level in the category hierarchy are associated with the catalog root through the catalog category association component.

Category
A category is a component of a catalog that represents a portion of the classification defined by the categories and category hierarchy in the catalog. You can associate a category to a catalog through the catalog category association. Both the shared category and the native category are associated thorough the catalog category association.

Catalog Category Association
Catalog category association represents the relationship between a catalog and a category, or a parent category and a child category. Each catalog category association represents one relationship between the catalog and a category or one relationship between a parent category and a child category.

Item Category Assignment
Item category assignment represents the assignment of the item to a category in a catalog. Each item category assignment represents the relationship between a category and an item.
**Item**

An item represents objects such as a product, service or template. An item is assigned through the item category assignment component.

**Attachment or Image**

Information is associated to the catalog and/or category, or both, through the attachment framework. Multiple attachments are supported but you can associate only a single attachment or attachment type image with a catalog or category.

**Catalog Formatting: Explained**

The format of a catalog is defined at the time the catalog is created and controls the behavior of the catalog at runtime.

When you format a catalog the layout controls three main areas and includes the following tasks, some fields are required, and others are optional.

- Catalog configuration
- Date enablement
- Category sharing

**Catalog Configuration**

You can configure the catalog, and this affects how the content behaves. The catalog configuration contains a set of attributes that define the catalog configuration. These attributes interact to define the runtime behavior of the catalog.

The configuration functions are:

- Catalog code: A unique identifier that is used.
- Catalog structure: The key flexfield structure used to define the catalog.
- Controlled at: Controls how items are assigned to categories and has two values. The first value is master level, which enables the automatic assignment of items to all child organizations associated with the master organization, if the current context is a master organization. The second value is organization level, which assigns the item only to the organization in the current context.
- Default category: Applies any time a new item is created. The newly created item is assigned to this category within the catalog automatically. The automatic assigned is controlled by the functional area.
- Catalog content: Controls what content can be added to the catalog and where the content can be added. This attribute has three values:
  - The Item at leaf levels allows items to be added only to the bottom level categories in the hierarchy.
  - The Items at all levels allows items to be assigned to any category in the hierarchy regardless of level.
  - Categories only allows categories to be added only to the catalog.
• Allow multiple item category assignment: When this option is selected, you can assign an item to one or more categories in the catalog. The default is deselected, which means that each item can be assigned to only one category in the catalog.

• Enable hierarchies for categories: When this option is selected, you can create a hierarchy for the catalog. The default is deselected, which means that the catalog cannot have a hierarchy and categories are associated with the catalog root.

• Enable automatic assignment of categories: When this option is selected, the catalog is built by automatically associating all categories, based on matching the catalog structure value to the category structure value.

Catalog Date Enablement
The date enablement function controls when the catalog is in an active state or inactive state by using the start date and end date attributes.

Category Sharing
The category sharing function enables sharing by reference to categories from a designated source catalog.

The sharing function has these attributes:

• Source catalog: A catalog that does not have sharing enabled from which categories, category hierarchies, and assigned items can be added to the catalog.

• Sharing content: Controls what content can be added from the source catalog. This attribute has three values:
  • Categories only: Only categories without assigned items can be shared.
  • Items only: Only categories with assigned items can be shared.
  • Items and categories: All categories can be shared.

Catalog Details: Explained

You can change a default category so that you can use it for item creation, or modify the inactive date so that the category is no longer used as you update a catalog. You can correct mistakes or reclassify the category due to shifting relationships within the category hierarchy.

You can view and edit a catalog on the Edit Catalog page when you have editing rights. For users that do not have rights to edit, the page is in read only mode.

The following aspects are important regarding managing and editing catalog details:

• Catalog header region
• Catalog detail tab
• Category hierarchy tab

Catalog Header Region
This region contains the catalog name and description, the selection of the default category and the start and end date for the catalog.
Catalog Detail Tab

The Detail tab contains:

- The configuration attributes for the catalog that controls the runtime behavior for the catalog.
- The sharing attributes for the catalog which controls the source catalog that will be used for sharing from and what content can be shared.
- The additional information which contains the descriptive flexfields defined for the catalog.

Category Hierarchy Tab

This contains the category hierarchy region in which the category hierarchy can be created and maintained. In addition, items can be assigned, and the usage of the category in other catalog can be viewed, and the attributes for the category and catalog category association can be edited.

Automatic Assignment Catalogs: Explained

The automatic assignment catalog feature enables you to reduce the cost of creating and maintaining a catalog. It is a simple way to create a nonhierarchical catalog because you do not have to add categories manually to the catalog.

All categories that have the same category structure value as the catalog are automatically assigned and associated to the catalog when you create a catalog category association for each category. Note that if you create a category in another catalog with the same structure value as the automatic assignment catalog, the category is added to your catalog. The categories displayed for auto assignment catalogs are refreshed only at startup and after you save.

Automatic Assignments

The automatic assignment feature is enabled during catalog creation when you select the **Enable automatic assignment of category** check box. When you open a new catalog, any categories that have the same category structure value as the catalog structure value for the catalog are automatically assigned to the catalog.

For example, Purchasing may maintain a master catalog containing all categories that represent commodities. Each commodity team can create categories for their commodity in their own catalog.

- The master catalog for Purchasing is named Purchasing and is configured during creation to support the automatic assignment of categories.
- The Electronic commodity team creates a catalog named Electronics and proceeds to create categories that represent the classification of their commodity. The Electronic commodity team creates the categories televisions, computers, and home theaters.
- The other commodity teams create and maintain separate catalogs.
- Because you enabled automatic assignments for the Purchasing catalog, any categories created by the commodity teams are added to the catalog automatically. The Purchasing managers can view the collection of all commodities represented as categories in the Purchasing catalog.
Manage Catalogs

Catalog Edits: Explained

The Edit Catalog dialog is a shared page that has two modes, view and update. The view mode displays the selected catalog in a read-only file. The update mode displays the selected catalog in an editable file. You must have edit catalog privileges to access the catalog in update mode. You can edit only an active or future-dated catalog.

The following fields are editable in the catalog:

- Catalog Name
- Description
- Start Date
- End Date
- Default Category
- Allow multiple item category assignment
- Addition Information
- Category Hierarchy
- Category Details
- Items assigned to category

Default Category

You can edit this field to select another category as the default category for item creation. You cannot remove the default category if the catalog is assigned to a functional area that requires a default category to be specified.

Allow Multiple Item Category Assignment

This check box is editable only until you assign an item to a category in the catalog.

Addition Information

You can edit the values of the descriptive flexfields attributes.

After you make changes, clicking the Save button saves the changes to the database but will not close the Edit Catalog page. Clicking the Save and Close button saves the changes to the database and closes the Edit Catalog page.

Categories and Catalog Relationships: Explained

Catalogs are used to organize and classify collections of items by associating categories to the catalog. The categories are organized to form a taxonomy and
items are assigned to the categories. When a category is associated with the catalog a catalog category association is created which specifies the relationship of the association. The catalog category association may also represent the relationship between two categories, for example a relationship between a parent category and a child category.

The following aspect is important regarding catalog category association:

- Date enablement attribute value

**Catalog Category Association**

The catalog category association is date enabled providing the control of when the catalog category association is active in the catalog and when the catalog category association is inactive. The catalog category association has two attributes to support date enablement; the start date and the end date. The start date is value is the first day that the catalog category association is available or active for use and the end date is the last day the catalog category association can be used, after this date the catalog category association is inactive. The date enablement attribute values are also used to control the visibility of content and the behavior of the category in the catalog. If a category association is inactive or end dated, having the value of the end date attribute past the current date, then the items cannot be assigned to the category.

A catalog category association will be set to inactive state when the category referenced by the catalog category association is set to an inactive state automatically, but the display will not be refreshed automatically.

**Date Enablement for Catalogs and Categories: Explained**

The catalog, categories, and catalog category association use date enablement to determine if the object specified is active or inactive based on the start date and end date. The following are date enablement definitions:

- **Active** An object is active when the current date is later than or equal to the value of the start date, but earlier than or equal to value of the end date.
- **Inactive** An object is inactive when the current date is later than the value of the end date.
- **Future dated** An object is future dated when the current date is earlier than the value of the start date.

You set the date enablement attributes are used to determine when a catalog, category, or catalog category association is used or visible.

- On the Manage Catalog page, a table filter determines which catalogs appear. The default value for the choice list is **Active**, indicating that only active catalogs will be displayed. You can select the value **All** to view both active and inactive catalogs.

- On the Edit Catalog page, on the category hierarchy tab, two table filters determine what categories and catalog category associations appear. The default values for the two choice lists are **Active**, indicating that only active categories and active catalog category associations will be displayed. You can select the value **All** to view both active and inactive categories and catalog categories associations.
• Other applications also use the date enablement attributes to filter information retrieved through application programming interfaces or services for catalogs.

The following illustration provides the date enablement attributes for these objects. The catalog, category, or the catalog category association has an internal state that is active or inactive.

The following aspects are important regarding date enablement for catalogs and categories:

• Start date
• End date
• Catalog and category objects
• Catalog category association
• Catalog and category rules

Start Date

The start date is defined as the first date that the object can be active. The start date can be future dated by setting the value to a date later than the current date. The start date value defaults to the system date if no date is entered during catalog or category creation.
End Date

The end date is defined as the last date that the object can be active. The object is end dated one second after the date specified by the value of **End Date**, that is the next day at 12:00:01 a.m. You cannot set the end date in the past. Also, you can change the end date from a condition when the object is ended to a new end date greater than or equal to the system date, causing the object to go from inactive to active. The end date value is optional during catalog or category creation.

Catalog and Category Objects

The start and end dates have been added for the catalog and catalog category association. The inactive date for categories has been renamed as the end date and the start date has been added.

Catalog Category Association

The catalog category association is used to specify the parent and child relationships between catalogs and categories and for category to category relationships. The catalog category association date enablement is independent of the category data enablement, except for the case where the category is end dated; the association is ended automatically as well. The catalog category association dates represents the state of the category for the catalog in which the category is associated.

Catalog and Category Rules

When a catalog is inactive the following rules apply:

- All operations for the catalog are disabled; the catalog is not editable.
- The catalog cannot be used in other processes.
- The catalog can be viewed only if you set filters on the Manage Catalog page to a value of **All**, enabling you to view active and inactive catalogs.

When a category is inactive the following rules apply:

- All operations for the category are disabled; the category is not editable.
- The category cannot be added to other catalogs.
- The category can be viewed only if you set the filters on the Edit Catalog page to a value of **All**, enabling you to view active and inactive catalogs.
- The system sets the catalog category association for the inactive category to inactive.

When a catalog category association is inactive the following rules apply:

- The category may be inactive or active; if the category is active it can be edited.
- The catalog category associations and related category can be viewed only if you set the association filter on the Edit Catalog page to a value of **All**, enabling you to view active and inactive catalogs.

When a catalog is future dated the following rules apply:

- All the operations of the catalog are enabled and the catalog is editable.
The catalog can be used in other processes, if allowed.
The catalog can be viewed only if the you set the filters on the Manage Catalog page to value of All.

Catalog Hierarchies: How They Fit Together

You use catalogs to organize and classify collections of items by associating categories with the catalog. You organize the categories to form a taxonomy and assign items to the categories. When you associate a category with the catalog, a catalog category association is created which specifies the relationship of the association. The catalog category association may also represent the relationship between two categories, for example, a relationship between a parent category and a child category.

The following diagram shows the relationships of the category hierarchy components:

Components

The components of a category hierarchy are:

- Catalog root: The topmost node in category hierarchy that represents the object called catalog.
- Category: The catalog component that is used to represent the classification structure.
• Catalog category association: The line in the diagram represents the relationship between a catalog and category or between a parent category and child category.

• Item category assignment: The dotted line in the dialog represents the relationship between a category and an item.

• Reference category: The category C5 in this diagram is shared as a reference category from a source catalog.

• Leaf level category: The lowest or bottom-level category in a category hierarchy. You can assign items to all levels in a category hierarchy if you configure the catalog to support this.

• Browsing category: The category C2 in this diagram is a browsing category. Browsing categories are categories that you add to the category hierarchy for the purpose of classification and do not have items assigned to them.

The category hierarchy does not have a limit on how many levels can be represented. The category hierarchy can have multiple hierarchies within a single category hierarchy.

**Category Edits: Explained**

Categories can be edited only from within an Edit Catalog page, the category hierarchy tab. The category can be edited by selecting row for the category in the category hierarchy table and editing the category information in the category detail panel. The category can only be edited if the category is active and the catalog is active or future dated. The category information can be edited in both the details and items tabs.

**Details and Items Tabs**

The following fields are editable in the category:

- Category name
- Description
- Attachments
- Category start date
- Category end date
- Items assigned to category

After changes are made the Save button will save the changes to the database but will not close the Edit Catalog page. The Save and Close button will save the changes to the database and close the Edit Catalog page.

**Catalog Category Association: Explained**

The catalog category association can be edited only within the Edit Catalog page, in the category hierarchy tab. The catalog category association start date and end date attributes can be edited in the details region.

**Category Catalog Associations**

You select the category in the category hierarchy table for the catalog category association that is being edited, the category details are displayed in the right
hand panel. The association start date and association end date are the only editable fields.

After completing the edits, click on the **Save** button to save your changes to the database, the Edit Catalog page will not close. The **Save and Close** button will save the changes to the database and close the Edit Catalog page.

**Category Details: Explained**

You can update category details when you select the row with the category in the category hierarchy table, the category details are displayed in the right hand panel in the user interface in an edit mode for all native categories. The category detail region contains information about the category that is associated to the catalog. It also contains the association start and end dates.

You can view and edit a catalog on the category details tab when you have editing rights. For users that do not have rights to edit, the page is in read only mode.

The following aspects are important regarding managing and editing category details:

- Category details tab
- Items tab
- Where used tab

**Category Details Tab**

The details tab contains information about the category that has been associated to the catalog. This information appears in all catalogs, since a category can be associated to one or more catalogs. The details tab contains the category configuration, category date enablement, association date enablement, and the additional attributes for the category.

The details tab contains attributes that define a category. Unstructured information is added through attachments. Images are added to a category and are displayed in the category details tab.

**Items Tab**

The item assignments are specific to the catalog where the category is associated.

**Where Used Tab**

The Where used tab contains a list of catalogs that the category is associated with.

**Categories: Explained**

You can create categories only in the context of a catalog, on the Edit Catalog page, Category hierarchy tab. When you select the **Create** icon in the category hierarchy table, it launches the Create Category dialog.

Consider the following important aspects when creating categories for catalogs:

- Create category region
- Configuration region
- Date enablement region
- Additional information region
Create Category Region

Enter a name and a meaningful description of the category in the create category region. Optionally, you can add an image and an attachment to this category.

Configuration Region

The key flexfield is determined during creation based on the catalog structure of the catalog. Enter the key flexfield segment values for the category. The number of key flexfield segment values depends on how you define the key flexfield at setup time. The category structure is the key flexfield structure instance that you create as part of the setup. When you define the key flexfield structure instance, you define the segments for the structure instance. For example, the family group and class group are segments. The segments appear in the Create Category dialog based on the key flexfield structure instance that you select.

The default value of the category content selection value is **Items and Categories**, but you can change the value. The values in the category content choice list vary based on the catalog content value.

The category content attribute value controls the content that you can add to this category.

- **Items Only**: Select to add only items to the category
- **Categories Only**: Select to add only categories to the category
- **Items and Categories**: Select to add both items and categories to the category

Date Enablement Region

Date enablement determines if an object is active or inactive based on the start date and end date. When categories are created, the default start date value is the current date. You can move the category start date beyond the current date to a future date within the category. The end date value is optional.

Additional Information Region

The additional information region contains all descriptive flexfield attributes that you set up for categories. You can edit the values of the descriptive flexfield attributes at the time of category creation.

After you complete the required fields for the catalog, clicking **OK** creates the category in the database, adds the category to the point of selection in the category hierarchy, and closes the dialog.

Category Moves: Explained

You use the move category function in the category tree table region of the Edit Catalog page. This is a table row action. The dialog is launched when you select an active or future dated category within the catalog and select this action. The move category function is disabled when the **Enable hierarchies for categories** check box is not checked or left unchecked.

Consider the following important aspects when moving categories within catalogs:
• Indentifying the new parent

**Indentifying the New Parent**

The dialog provides the current category parent and allows you to pick a new category parent. Only the legal category parents are displayed in the choice list. The category list within the **New Parent** choice list is filtered by based on a set of rules:

• The new parent category must be an active or future dated category; the end date value of the category must be later than the current system date.
• The value of the category content for the new parent category must allow the selected category to be added; the legal values are items and categories and categories only.
• A selected category associated with the catalog at a level below the categories at the root categories can be moved to the root of the catalog.
• The new parent category catalog category association must be active; the end date value of the catalog category association must be later than the current system date.

**Import Category Hierarchies: Explained**

Category hierarchy can be created and maintained through a spreadsheet interface reducing the amount of time to create and maintain catalogs. Existing catalog content can be exported and the content used in other catalogs for catalog category hierarchies. The following aspects are important regarding category hierarchy import used in catalogs:

• Spreadsheet interface
• Export category hierarchy

**Spreadsheet Interface**

You can manage the catalog category hierarchy to use the spreadsheet interface that is available in the Edit Catalog page by using the **Export Hierarchy** button to download existing catalog content, modify this content in a spreadsheet, and upload the content back into the Product Information Management application.

**Export Category Hierarchy**

You use export category hierarchy for example, when you need to provide the category hierarchy to a partner. Your partner has the capability to import the catalog file using an Excel spreadsheet.

You can export the category hierarchy from our catalog and it can be used by partners. If your partner has the Oracle Product Information Management solution, they can directly import the category hierarchy into their catalog.

**Catalog or Category Attachments: Explained**

Catalogs and categories support attachments and use a common component for managing attachment content. You can add attachments on both the Create Catalog and Edit Catalog pages.
The attachment component displays a green plus sign icon indicating that no attachments are available for the object. The Attachment dialog appears when you click the green plus sign icon. You define the attachment by selecting the attachment type, file name or Uniform Resource Locator (URL), title, description, and by indicating whether the attachment can be shared with other objects. Once you define the attachments and click the **OK** button, that attachment title appears in the attachment component region of the page along with a red X icon that you can click to delete the attachment.

The attachment file types are:

- File
- Repository File/Folder
- Text
- URL

**File**

You must provide a title for the file and create a description for the attachment. You select a file to upload from your desktop.

**Repository File/Folder**

You click the **Browse** button to attach a repository file/folder from the document repository to a catalog. The attachment repository contains existing attachments and is organized as a set of folders. The **Browse** button launches the Attachment Repository dialog to enable you to select an attachment. You must provide a title for the repository file/folder and create a description for the attachment.

**Text**

Enter the text string in the field that you want to appear as an attachment. You must provide a title for the text and create a description for the text attachment.

**URL**

Enter the URL address to a web page that you want to attach to the catalog. You must provide a title for the URL attachment and create a description for it.

The **Share** check box alerts users that you added an attachment and the date that you performed the task.

**Items to Categories Assignment: Explained**

You can assign items to categories on the Edit Catalog page, category hierarchy tab, on the category detail item tab. You can assign items only to active categories and categories where the **Category Content** field value is **Items and Categories** or **Items Only**. In addition, you can configure catalogs to control item assignment to categories within the catalog by selecting the **Allow multiple item category assignment** check box, which allows items to be added to all levels of the category hierarchy.

You select items from a choice list and add them to the category. The choice list is filtered based on a set of rules:

- Item data level security: Displays only the items that the user has permission to view and assign.
• Organization context: Based on the organization context that is controlled by a choice list in the item table header, only the items assigned to organizations are displayed.

Controlling Item Assignment

You also control item assignment by selecting the value of the Controlled at check box. If you select the Master Level value and the organization context is a master organization, the items are automatically assigned to all child organizations that are associated with the master organization.

Catalog Publishing: Explained

Other applications can use catalog data if you export the catalog content. For example, you may want to export catalog content to use as a monthly report of all items assigned to a specific catalog. You can use the default publish template provided in hyper text markup language (HTML). You can specify the content and layout of the catalog information. When the catalog is published, you select the format and initiate the creation of the content in the file.

The following aspects are important regarding catalog data to be published:

- Publish a catalog
- Type of catalog content that can be published

Publish a Catalog

You initiate a search for a catalog from the Manage Catalogs page, select the row corresponding to the catalog that you want to publish and select the Publish action. The application generates the report based on the default template in HTML format, and the locale prior to creation of the file. You can select a new template or format from the report window. The content displayed for items, categories, catalog categories, and catalog is based on the publish template.

Type of Catalog Content That Can Be Published

The default catalog publish template allows the publication of the catalog header details, category hierarchy, category details, and category item assignments. The order of a published report begins with the catalog header and the catalog category details. If the category has a child relationship then the catalog category association details for the child category follows. If the child category has a hierarchy, then the complete hierarchy under the category is published with the catalog category association details and categories details.

FAQs for Manage Catalogs

How can I share catalog content?

Categories can be shared across multiple catalogs allowing catalog content to be reused and saving the work needed to maintain multiple copies of the categories.
In the case of category sharing, the category structure in the source catalog can be different than the native catalog.

Categories can be shared using two methods; the first method is directly associating the category to the catalog. The category is added to the catalog and can be edited in the catalog or any catalog the category is associated to. The items assigned to the category are not shared, but are assigned to the category in context with the catalog the category is associated. For example if the category name or description is changed in one catalog, the change will be reflected in all catalogs where the category is associated, but if items are assigned to a category, the assignment will be for that single catalog.

The second method of sharing categories is adding a category by reference into the catalog. During the creation of the catalog, sharing can be enabled by specifying a single source catalog that will be used for sharing by reference and setting the value of the sharing content to control what content will be shared from the source catalog. The advantage of using sharing by reference is source catalog content can be shared to multiple catalogs and maintained in a single place, the source catalog. In addition, the referenced content can be more than one category, for example a complete category hierarchy and any assigned items to categories in shared content can also be reference within the catalog.

**How can I define category hierarchies?**

Categories can be organized to represent classification taxonomies. The hierarchy organizations for categories have parent and child relationships that form a tree structure. The category hierarchy is created and maintained within the Edit Catalog page, category hierarchy tab. The category hierarchy is shown in true relationship to the way it is defined.

The category hierarchy can be created using two methods: the first is manually creating the hierarchy by adding referenced categories, duplicating categories or creating category for the catalog.

The second method for creating the hierarchy is by importing the category hierarchy through the spreadsheet interface. The category hierarchy can be exported from other catalog or other sources, edited and imported into a new catalog, additionally it can be added manually to the spreadsheet.

The category hierarchy can be edited using **Move Category**. The catalog category association cannot be deleted, but can be end dated to make the catalog category association inactive. The category hierarchy table provides a choice list filter that controls what catalog category associations and categories area displayed based on the date enablement. The category hierarchy can also be edited by exporting the complete hierarchy, editing it and importing the category hierarchy back into the catalog.

**How can I duplicate categories?**

You can select and duplicate a category as a quick way to create a similar category configuration. Selecting the **Duplicate** icon action launches a Create Category dialog that has attribute fields populated based on the selected category attribute values. The category name is prefixed with **Copy_** followed by the name of the selected category. You fill in the required field information in the key flexfield segment values which are blank. Once the category attributes are updated and the key flexfield segments values are entered, the **OK** button adds
the newly created category into the category hierarchy of the selected category you have configured.

**How can I add categories?**

Categories are catalog components that are associated to a catalog for purpose of classification of items. You can add existing categories to the point of selection which can be a category in the hierarchy or the root of the catalog. If no category is selected, the default is the root of the catalog.

You can add categories by selecting the **Add Category** field and selecting the value **Add Category**. You can then search for existing categories based on the value of the catalog structure for the catalog. You can narrow the search for existing categories by using the **Advance Search** region in the dialog. You can add each selected category by selecting the **Apply** button and the add category region remains open. The **OK** button adds a category if a category is selected and then closes the dialog.

**How can I add shared categories?**

Adding a shared category is similar to adding an existing category except the category is selected from the catalog that has been designated as a source catalog. The sharing content attribute value determines what content is shared from the source catalog. A category within a source catalog that has been added to a native catalog is also known as a referenced category. You use the drop list menu from the Add Categories menu, and the Shared Category option will be disabled if the catalog has not been configured for category sharing.

**How can I add images to a catalog or category?**

You can attach an image from your desktop or from a configured repository to a catalog or a category, or both. The image is displayed in the catalog detail and the category detail section of the catalog page. Only one image can be associated with a catalog or category. To attach an image, select the green plus icon to launch the Manage Attachment dialog. The image attachment type can have values of **File** or **Repository File/Folder** and is selected in this dialog. The title you provide for the image attachment will appear under the image that is displayed in the catalog. The description you provide is not displayed. **Browse** will allow you to select the file to be used as the image for the catalog or category. After the information is entered in to the dialog, you click the **OK** button to load the image and the image attachment title will be displayed under the image. The image will not initially be displayed until the catalog is saved. The image can be replaced with another image by selecting the red X to delete the existing image and entering a new image.

**What is catalog mapping?**

You use **Catalog Category** mapping to map categories of different catalogs to the reporting categories in other catalogs. This feature allows one or more categories within a catalog to be mapped to category in a second catalog. For example, suppose that you want to roll up the costs associated with allow items assigned to a set of categories in catalog. Catalog mapping allows you to select a category in a catalog, and map all the categories in the set to that category. When you use...
this feature you are required to write code to do the roll up as identified in the example.

**How can I map default catalogs?**

You can map a catalog to be assigned to a functional area such as Purchasing. When a catalog is assigned to a functional area, the catalog will behave based on the rules you defined for that functional area. Only one catalog can be assigned to a functional area.

**Define Supplier Configuration**

**Supplier Numbering: Explained**

The Procurement Application Administrator is responsible for supplier numbering setup. Suppliers created through the Create Supplier task flow, through the supplier registration process, or supplier import are automatically numbered using a numeric sequence. The starting supplier number is defined in the Specify Supplier Numbering setup page (the seeded default number is 1). The supplier number then increments automatically as numbers are assigned during supplier creation.

Additionally, the next supplier number can be updated at any time, not just during initial setup, if for example there's ever a need to skip a range of supplier numbers. The application will validate that the number is not already used.

**Business Classifications: Explained**

Business classifications support the tracking of supplier certifications that are important to companies for different reasons, such as for supplier diversity programs. Since classification requirements can vary by country or region the seeded classifications can be changed. The application allows you to capture certification details for each classification such as Certificate Number, Certifying Agency, and Expiration Date.

The following seeded classifications are provided for tracking suppliers.

- Hub Zone
- Minority Owned
- Subclassification: (African American, American Indian, Asian, Hispanic)
- Service-disabled Veteran Owned
- Small Business
- Veteran Owned
- Woman Owned

The Business Classifications subtab displays an alert if there are certificates that are expired or about to expire. The alert is only raised if a certificate expired in the past 30 days or about to expire in the next 30 days.
Define Transaction Taxes

Define Transaction Taxes: Overview

Oracle Fusion Tax provides a single-point solution for managing your transaction-based tax requirements. In the Define Transaction Taxes activity, set up your entire tax configuration.

Oracle Fusion Tax:

- Uniformly delivers tax services to all Oracle Fusion application business flows through one application interface
- Provides a single integration point for third-party tax products and services
- Is configurable and scalable for adding and maintaining country-specific tax content

With Oracle Fusion Tax, you can model your taxes according to the needs of the following local and international tax requirements:

- Both simple and complex country-specific tax legislation
- Cross-border transactions, including exports and Intra-European Community transactions
- Intercompany transactions
- Local compliance requirements for recording and reporting
- Continual changes to tax legislation, such as new taxes, local law changes, special tax rates, and special exceptions for products and customers

You can manage the entire configuration and maintenance of tax content from the one Oracle Fusion Tax application. Using one application ensures a uniform tax setup across applications, with a centrally managed system of automated tax services and control over manual intervention and update.

Task Lists

The Define Transaction Taxes activity is logically defined with prerequisite tasks, core tax configuration tasks, optional setup tasks, and validate configuration tasks. The activity categories include:

- Define Tax Geographies: Configure tax geographies to define geographical regions that share the same tax requirement. These prerequisite tasks are required for core tax configuration but they might not have been defined in the previous steps of the Financials offering.
- Define Tax Regimes: Configure tax regimes for the taxes in each country and geographic region where a separate tax applies. These tasks are most commonly used by all the implementations. You should be able to calculate taxes on the transactions based on this configuration.
- Define First Party Tax Profiles: Configure tax profile details that control the transaction tax activities for your first party legal entities, legal reporting units, and business units.
• Define Third Party Tax Profiles: Configure tax profile details that control the transaction tax activities for your third party customer, customer sites, supplier, and supplier sites.

• Define Occasional Implementation Setups: Configure initial tax setup that impacts tax calculation and reporting. These tasks either are predefined and you do not have to configure them unless the predefined data needs to be extended or these are tasks required only for certain implementations.

• Verify Tax Configuration: Verify the transaction tax configuration by simulating transaction data and reviewing tax calculation results.

**Defining Transaction Taxes: Critical Choices**

With Oracle Fusion Tax, you can model your tax requirements according to the needs of local and international tax requirements. These requirements include:

• Both simple and complex country-specific tax legislation

• Cross-border transactions

• Local compliance requirements for recording and reporting

• Continual changes to tax legislation, such as new taxes, local law changes, special tax rates, and special exceptions for products and customers

In order to determine how to set up your tax configuration, you must first analyze your tax requirements.

**Analyzing Your Tax Requirements**

The following table represents key decisions that you must make when you analyze your tax requirements and use Oracle Fusion Tax and other Oracle Fusion applications to implement a solution.

<table>
<thead>
<tr>
<th>Question</th>
<th>Consideration</th>
<th>Impact to Tax Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who am I?</td>
<td>You must first answer questions about yourself and your relationship to the legal and regulatory agencies that enable you to operate in one or more counties.</td>
<td></td>
</tr>
<tr>
<td>Where do I have operations and businesses?</td>
<td>Identify the countries in which you operate. You will need to identify the country where you are legally registered and the countries where you have subsidiary companies that are legally registered or have a legal presence.</td>
<td>Use Oracle Fusion Legal Entity Configurator to capture information about your legal entities and legal registration.</td>
</tr>
<tr>
<td>What taxes am I subject to?</td>
<td>Analyze your tax environment for each of the countries in which you operate.</td>
<td>Set up your tax regimes, taxes, and tax jurisdictions according to the tax requirements for each country.</td>
</tr>
</tbody>
</table>
| What are the operations and businesses that I have? | Consider the types of operations and businesses in which you are engaged and the countries where you have legal entities or reporting units. The type of industries that you work under (for example, mining, telecommunications, and pharmaceuticals), the kind of operations in which you engage (for example, trading, manufacturing, and services), and the scale of your operations (for example, your turnover, company size, and growth) may all impact your taxability. | Use the classifications feature to categorize or classify your first parties under various classification schemes. In analyzing your operations, you can associate the three main classifications of a transaction to:  
- What you do: Use transaction fiscal classifications.  
- What products you buy or sell: Use product fiscal classifications.  
- Who your customers and suppliers are: Use party fiscal classifications. |
| What do I do? | Identify and classify the transactions that you enter into. For example, do you primarily sell physical goods? If you do, do you manufacture them, or do you buy and sell them without additional manufacturing? Do you sell these goods in another state or province? Do you export these goods? Do you provide or use services? | Use Oracle Fusion Tax to create fiscal classifications to classify and categorize your transactions in a common manner across your organization. Use these fiscal classifications in tax rules to obtain the appropriate tax result. |
| What products do I buy or sell? | Determine the products that you buy and sell as they impact the taxes to which you are subject. For example, you must register for, and therefore collect and remit, service taxes only if you provide taxable services. If you manufacture goods for export, you may not be subject to taxes on the purchases that go into the manufacture of such goods. | Where Oracle Fusion Inventory is installed use the Inventory Catalog feature with Oracle Fusion Tax product fiscal classifications and intended use functionality to classify the taxable nature and intended use of the items. You can then define tax rules using these classifications to obtain the appropriate tax result. Define product category and noninventory-based intended use fiscal classifications to address classification needs for transactions that do not use inventory items. |
Who are my customers and suppliers?

Determine the types of customers and suppliers with whom you do business, as they can impact the taxes to which you are subject or the tax status or tax rate that applies. For example, let’s say that you are a company in the UK that supplies physical goods to another country that is also a member of the European Union. The transaction rate for UK VAT is dependant on whether the customer is registered for VAT in the country to which the supply is made.

Use the party classifications feature to categorize or classify your customers and suppliers. You can use these classifications in your tax rules to derive the appropriate tax result.

You create a party fiscal classification by assigning an Oracle Fusion Trading Community Model class category to a party fiscal classification type code that you define. The Trading Community Model class codes defined under the class category become fiscal classification codes belonging to the party fiscal classification type. You can create a hierarchy of party fiscal classification types to reflect the levels of codes and subcodes within the Trading Community Model classification.

**Scope Values for Define Transaction Taxes Task List: Explained**

The purpose of scope is to define the parameters of your implementation project by setting the context of a task list during initial configuration.

The foundation tax setup is an incremental setup where each step of the foundation configuration builds on the previous step. The task list is organized sequentially to ensure that you perform setup tasks in the order required. You can define scope values at incremental steps in the implementation project to pass to subsequent tasks to ensure:

- Continuity
- Ease of setup

When exporting setup data based on setup migration services, the scope values serve as parameters to control the data selected for export to the respective configuration package. Scope is a valuable tool when implementing, but tax scope values are not a required element of the implementation and you do not need to define them.

**Defining Scope**

When implementing transaction tax, you can define scope values for taxes, tax jurisdictions, tax statuses, tax rates, and tax recovery rates in the foundation setup. To set scope, you can:

- Select and add multiple values
- Create a new value

When you select the scope value, that value defines the context of that setup. For example, if you select a tax regime to use as a scope value for a tax, that value is automatically populated in the search attributes on the Manage Tax page. That
tax regime’s attributes are also populated in the Create Tax page. The same logic applies to the next step in the foundation setup.

**Scope Values**

The following table identifies where you define the scope value in the Define Transaction Taxes task list:

<table>
<thead>
<tr>
<th>Where Scope is Defined</th>
<th>Scope Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage Taxes</td>
<td>Tax regime</td>
</tr>
<tr>
<td>Manage Tax Jurisdictions</td>
<td>• Tax regime</td>
</tr>
<tr>
<td>Manage Tax Statuses</td>
<td>• Tax</td>
</tr>
<tr>
<td>Manage Tax Rates</td>
<td>• Tax regime</td>
</tr>
<tr>
<td>Manage Tax Rates</td>
<td>• Tax</td>
</tr>
<tr>
<td>Manage Tax Recovery Rates</td>
<td>• Tax regime</td>
</tr>
<tr>
<td>Manage Tax Recovery Rates</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Tax Rate Determination Rules</td>
<td>• Tax regime</td>
</tr>
<tr>
<td>• Manage Direct Rate Determination Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Account-Based Direct Tax Rate Determination Rules</td>
<td></td>
</tr>
<tr>
<td>• Manage Tax Classification-Based Direct Tax Rate Determination Rules</td>
<td></td>
</tr>
<tr>
<td>• Manage Tax Applicability Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Place of Supply Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Tax Registration Determination Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Tax Status Determination Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Taxable Basis Formulas</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Taxable Basis Determination Rules</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Tax Calculation Formulas</td>
<td>• Tax</td>
</tr>
<tr>
<td>• Manage Tax Calculation Rules</td>
<td>• Tax</td>
</tr>
</tbody>
</table>

**Foundation Tax Configuration: Points to Consider**

Use Oracle Fusion Tax to set up and maintain your transaction tax requirements in all geographic locations where you do business. Foundation tax configuration refers to a set of tax setup components that you will use to satisfy your tax
requirements. At transaction time, Oracle Fusion Tax uses your tax configuration to determine the taxes that apply to each transaction and to calculate the tax amounts.

Foundation tax configuration components consist of:

- Tax regimes
- Taxes
- Tax jurisdictions
- Tax statuses
- Tax rates

## Foundation Tax Configuration

Complete the setup tasks to create a basic tax configuration for each of your tax regimes. A foundation tax configuration contains the data applicable to the taxes belonging to a tax regime. The following table describes the appropriate levels of specifying setup options for foundation tax components and provides a Canada Goods and Services Tax (GST) and Harmonized Sales Tax (HST) example for each component.

<table>
<thead>
<tr>
<th>Component</th>
<th>Appropriate Level to:</th>
<th>Typically, Not Appropriate Level to:</th>
<th>Canada GST and HST Example</th>
</tr>
</thead>
</table>
| Tax Regime | • Share tax content among legal entities and business units.  
• Enable partner integration.  
• Associate fiscal classifications.  
• Define tax reporting types and codes.  
• Define features to influence setup task list. | • Define configuration owner tax options.  
• Define application tax options.  
• Define party tax profiles. | CA GST & HST |
| Tax       | • Enable controls to influence tax behavior.  
• Specify defaults that are commonly applicable.  
• Define applicability tax rules.  
• Define customer exemptions.  
• Specify party registrations. | • Share tax content.  
• Define integration with partners. | CA GST  
CA HST |
### Advanced Tax Configuration: Points to Consider

Create a simple tax model using tax rule defaults that you define in setting up your foundation tax configuration. You can also create tax rules for your complex tax requirements that consider each tax requirement related to a transaction before making the final tax calculation. When running the tax determination process, Oracle Fusion Tax evaluates, in order of priority, the tax rules that you have defined against the foundation tax configuration setup and the details on the transactions. If the first rule is successfully evaluated, the result associated with the rule is used. If that tax rule is not successful, the next rule is evaluated until either a successful evaluation or a default value is found.

Advanced tax configuration consists of tax rules to define exceptions to the default results.

### Advanced Tax Configuration

The complexity of tax rule setup falls into three general categories: no tax rules required, simple tax rule regimes, and complex tax regimes. This table presents the scenarios and actions associated with each of these categories.
<table>
<thead>
<tr>
<th>Category</th>
<th>Scenario</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>No tax rules required</td>
<td>The tax authority levies tax on all sales and purchase transactions at the same rate. Neither tax applicability nor the tax rates and recovery rates vary by the parties to the transaction, the products or services in the transaction, or the business processes involved in the transaction.</td>
<td>For the tax, define tax rule defaults for the tax status, tax rate, and tax recovery rate. The tax determination process uses the tax rule defaults to determine the tax.</td>
</tr>
<tr>
<td>Simple tax rule regimes</td>
<td>The tax authority levies tax on your transactions at the same rate, with a simple set of identifiable exceptions. The exceptions either apply to one part of the transaction only, such as to certain parties, or to a combination of parties, products, and transaction processes that you can summarize in a simple way.</td>
<td>Create a simple set of rules, for example, to identify place of supply and tax registration, and use the tax rule default values for the other processes. The tax determination process uses the tax rules and the tax rule defaults to determine the tax.</td>
</tr>
<tr>
<td>Complex tax regimes</td>
<td>Tax regimes in certain countries require a complex logic to determine the applicable taxes and rates on a transaction. Both tax applicability and tax rates can vary, for example, by place of origin and place of destination, party registration, tax status, service, or a combination of factors. In some cases, the taxable amount of one tax may depend upon the amount of another tax on the same transaction. And in rare cases, the tax amount itself may depend on the tax amount of another tax.</td>
<td>Set up tax rule to define the logic necessary to identify each step of the tax determination process. The tax determination process uses the tax rules to determine the tax.</td>
</tr>
</tbody>
</table>

**Define Exception to Default Results**

Set a tax rule default value to the most commonly used value for tax determination. In the case of tax registration the default or most commonly used value for registration party is ship-from party. However, you can set up a rule to provide additional logic to use the registration of the bill-to party if the registration status is **Not Registered** for the ship-from party for purchase transactions. Create a determining factor set with the registration status and transaction business category determining factors along with condition sets to provide values for the respective determining factors.

For this example, the following setup exists for the Determine Tax Registration tax rule:

- Tax rule default: The default for tax registration is ship-from party.
- Tax rule: If the supplier is not registered, then you should consider the tax registration of the bill-to party.

When the following conditions are true, then the tax registration is the same as that defined for the bill-to party:
The tax determination process determines the tax registration by first considering the Determine Tax Registration tax rule and then the default party registration. As a result of this rule, the tax determination process determines that for a purchase transaction, if the supplier is not registered, the tax registration of the bill-to party is considered.

### Define Tax Geographies

#### Place Information: Explained

All tax regimes need information about place or geography. Information is required to determine:

- Where the tax is applicable
- The tax rules that can identify when a transaction is an export, or delivered to another country, or deliveries inside or outside an economic region such as, the European Community (EC).
- Specific regions such as, city, country, and states for US Sales and Use Tax or provinces in Canada.

To support these requirements, Oracle Fusion Tax allows you to define and use geography regions and tax zones. Geography regions and tax zones provide a conceptual model to use place information on transactions and information related to the transaction.

The following types of places are supported for tax purposes in Oracle Fusion Tax:

- Country information: Use country as a specific geography element in tax rules to define tax regimes, taxes, and tax jurisdictions.
- Geography elements: Use geography elements or levels defined in the Oracle Fusion Trading Community Model geography functionality in tax rules to define tax regimes, taxes, and tax jurisdictions.
- Tax zones: Use geography elements or levels defined in Trading Community Model geography in tax rules to define tax regimes, taxes, and tax jurisdictions.

Use place information for determining factors within tax rules in the tax determination process. Also, use place information while defining tax regimes, tax geography, and tax jurisdictions.

#### Country Information

Country is a required field in all of the tax-related address locations. The country fields are supported by a predefined ISO 3166 country name and two-character country code. For more information on country names and codes, see [http://www.iso.org/iso/english_country_names_and_code_elements](http://www.iso.org/iso/english_country_names_and_code_elements).
You do not set up a country as a specific geography level in Trading Community Model geography because country is an inherent part of all tax-related address locations.

---

**Tip**

Use the highest level of geography, typically country, wherever possible.

---

**Geography Elements**

Define geography elements as part of Trading Community Model geography. They control the use of geography and addresses throughout Oracle Fusion. Oracle Fusion Tax commonly uses the following features: geography or tax zones, geography levels, address controls, and geography name referencing.

Use geography levels to define the levels of geography that are used within a country. For example, addresses in the US comprise of state, county, city, street, and postal code. Addresses in the UK comprise of county, city or town, street, and postal code. There may be other geography elements as well, such as building. From a tax perspective it is only those elements of the address that are referenced for tax purposes. For example, state, county, and city are important for US Sales and Use Tax while county in UK is not relevant from a tax perspective and therefore, you do not need to set it up.

---

**Tip**

When address elements are needed for tax purposes, such as country and city for US Sales and Use Tax, set these address levels as mandatory within Trading Community Model geography. This ensures that these elements are always present on all applicable addresses.

Setting address levels as mandatory ensures that amended or newly applicable addresses are validated and that the level is either derived or entered. When you are setting up migrated addresses ensure that they are also compliant with the mandatory levels being present. This should be validated and any address levels added as part of the migration process.

---

The geography name referencing process within Trading Community Model geography links specific addresses to the levels defined in the geography setup. This process is typically automatic. However, when you encounter issues, you may need to trigger this process to ensure that all addresses are correctly linked to their applicable levels.

---

**Tax Zones**

Use the tax zone functionality when you need to identify a group of geography elements while calculating tax. Tax zones are defined as part of Trading Community Model geography.

For example, in the EC it is important to know whether goods and services are being delivered within the EC. Use the tax zone functionality to create a tax zone, which defines the membership to the EC as well as, the dates on which a country became the member.

---

**Tip**

Create a generic tax zone so that you create a tax zone type that can be used in multiple situations. For example, for a tax zone type needed to identify EC,
create a generic tax zone type for all economic communities, which can later be used in other situations where economic communities or trade agreements affect tax determination.

You can also use the tax zone functionality to group postal codes to provide useful groupings that can identify some higher-level tax regions such as, cities or counties.

**Country Information: How It Works in Tax Rules and on Transactions**

Geography determination factors allow you to use country information in the tax rules. A combination of determination factor class, class qualifier, and determining factor represent these determination factors. Specify the taxation country at transaction time which is used, along with the tax rules, during the tax determination process.

**Country Information in Tax Rules**

Use geography as the determining factor class, location type on the transaction as the class qualifier, and country as the determining factor. You can also use country as a tax rule qualifier.

The tax determining factors for locations are given generic names such as ship-to and bill-from, depending on the transaction types. The transaction types are **Order-to-cash**, for example, Oracle Fusion Order Management and Oracle Fusion Receivables, and **Procure-to-pay**, for example Oracle Fusion Purchasing and Oracle Fusion Payables.

Oracle Fusion Tax translates these generic locations into specific locations based on the transaction as shown in the following table:

<table>
<thead>
<tr>
<th>Generic Party</th>
<th>Order-to-Cash Party</th>
<th>Procure-to-Pay Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bill-from party</td>
<td>Location assigned to the business unit for the transactions</td>
<td>Supplier</td>
</tr>
<tr>
<td>Bill-to party</td>
<td>Customer</td>
<td>Location assigned to the business unit for the transactions</td>
</tr>
<tr>
<td>Ship-to party</td>
<td>Customer (ship-to) party site</td>
<td>Ship-to location on the line</td>
</tr>
<tr>
<td>Ship-from party</td>
<td>Warehouse on the line. If there is no warehouse on the line, such as with services, the default location assigned in the Receivables system parameters is used.</td>
<td>Supplier (ship-from) party site</td>
</tr>
<tr>
<td>Point of acceptance party</td>
<td>Customer point of acceptance party</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Point of origin party</td>
<td>Customer point of origin party</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

**Country Information at Transaction Time**

Specify the taxation country on the transaction to identify the country in which the transaction is deemed to have taken place for taxation purposes. The default value is the country of the legal entity. Use the country name to search for country defaults, which control the fiscal classification defaults, party tax profile defaults, and tax regime and tax defaults. Use the country name to select the following fiscal classifications associated with that specific country:
• User-defined fiscal classifications
• Product categories
• Intended use fiscal classifications
• Transaction business categories

Using Country Information in Tax Rules: Example

For many regimes, it is important to know if the supply of goods is exported. The easiest way of doing this is to ensure that the ship-from location is from the country in question and the ship-to location is a different country.

The following scenario illustrates setting up tax rule components to identify if the goods are exported from the United States.

Scenario

Use geography as the determining factor class, country as the class qualifier for ship-from and ship-to locations, and country as the determining factor as shown in the following table:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-from</td>
<td>Country</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>Country</td>
</tr>
</tbody>
</table>

Create a condition set that refers to this geography determining factor as follows:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-from</td>
<td>Country</td>
<td>Equal to</td>
<td>United States</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>Country</td>
<td>Not equal to</td>
<td>United States</td>
</tr>
</tbody>
</table>

Use this combination of determining factors in any situation where you need to identify exports from the United States.

Geography Elements: How They Work in Tax Rules

Geography determination factors allow you to use geography elements in tax rules. A combination of determination factor class, class qualifier, and determining factor represent these determination factors.

Geography Elements in Tax Rules

Use geography as the determining factor class, location type on the transaction as the class qualifier, and geography level such as county, province, or city, as the tax determining factor.

The tax determining factors for locations are given generic names such as ship-to and bill-from, depending on the transaction types. The transaction types are Order-to-cash, for example, Oracle Fusion Order Management and Oracle Fusion Receivables, and Procure-to-pay, for example Oracle Fusion Purchasing and Oracle Fusion Payables.
These generic locations are mapped to the specific location, based on the transaction as shown in the following table:

<table>
<thead>
<tr>
<th>Generic Party</th>
<th>Order-to-Cash Party</th>
<th>Procure-to-Pay Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bill-from party</td>
<td>First party legal entity</td>
<td>Supplier</td>
</tr>
<tr>
<td>Bill-to party</td>
<td>Customer</td>
<td>First party legal entity</td>
</tr>
<tr>
<td>Ship-to party</td>
<td>Customer (ship-to) party site</td>
<td>First party legal entity</td>
</tr>
<tr>
<td>Ship-from party</td>
<td>First party legal reporting unit</td>
<td>Supplier (ship-from) party site</td>
</tr>
<tr>
<td>Point of acceptance party</td>
<td>Customer point of acceptance party</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Point of origin party</td>
<td>Customer point of origin party</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

You can also use the geography level as a tax rule qualifier.

**Using Geography Levels in Tax Rules: Example**

Use the geography element in tax rules to identify a specific geography region when taxes in a specific country need to identify specific geography elements below the country level. For example, in US Sales and Use Tax for county taxes, there may be specific rules for a specific state.

The following scenario describes how you can set up tax rule components to identify when goods are being delivered to a specific state, such as Ohio.

**Scenario**

Use geography as the determining factor class, ship-to as the class qualifier, and state as the determining factor as shown in the following table:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>State</td>
</tr>
</tbody>
</table>

Create a condition set that refers to a specific state value as follows:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>State</td>
<td>Equal to</td>
<td>Ohio</td>
</tr>
</tbody>
</table>

You can use this combination of determining factors in any situation where you need to identify specific deliveries to a specific state.

**Tax Zones: How They Work in Tax Rules**

Geography determination factors allow you to use geography elements in the tax rules. A combination of determination factor class, class qualifier, and determining factor represent these determination factors.

**Tax Zones in Tax Rules**

Use geography as the determining factor class, location type on the transaction as the class qualifier, and tax zone type such as county, as the determining factor.
The tax determining factors for locations are given generic names such as ship-to and bill-from, depending on the transaction types. The transaction types are **Order-to-cash**, for example, Oracle Fusion Order Management and Oracle Fusion Receivables, and **Procure-to-pay**, for example Oracle Fusion Purchasing and Oracle Fusion Payables.

These generic locations are mapped to the specific location based on the transaction as shown in the following table:

<table>
<thead>
<tr>
<th>Generic Party</th>
<th>Order-to-Cash Party</th>
<th>Procure-to-Pay Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bill-from party</td>
<td>First party legal entity</td>
<td>Supplier</td>
</tr>
<tr>
<td>Bill-to party</td>
<td>Customer</td>
<td>First party legal entity</td>
</tr>
<tr>
<td>Ship-to party</td>
<td>Customer (ship-to) party site</td>
<td>First party legal entity</td>
</tr>
<tr>
<td>Ship-from party</td>
<td>First party legal reporting unit</td>
<td>Supplier (ship-from) party site</td>
</tr>
<tr>
<td>Point of acceptance party</td>
<td>Customer point of acceptance party</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Point of origin party</td>
<td>Customer point of origin party</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

You can also use tax zones as tax rule qualifiers.

**Using Tax Zones in Tax Rules: Example**

For the European Community (EC) or the Economic Union (EU) it is important to know whether goods and services are being delivered within the EC. Use the tax zone functionality to create a tax zone that defines the membership of the EC as well as the dates on which a country became a member.

The following scenario describes the use of a partial condition set that you can use within tax rules to define when a delivery is being made to an EC from the United Kingdom.

**Scenario**

Use geography as the determining factor class, ship-to as the class qualifier, and all economic communities and country as the determining factors of the tax zone type as shown in the following table:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>All Economic Communities</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>Country</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-from</td>
<td>Country</td>
</tr>
</tbody>
</table>

Create the condition set as follows:

<table>
<thead>
<tr>
<th>Determining Factor Class</th>
<th>Class Qualifier</th>
<th>Determining Factor Name</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>All Economic Communities</td>
<td>Equal to</td>
<td>European Community</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-to</td>
<td>Country</td>
<td>Not equal to</td>
<td>United Kingdom</td>
</tr>
<tr>
<td>Geography</td>
<td>Ship-from</td>
<td>Country</td>
<td>Equal to</td>
<td>United Kingdom</td>
</tr>
</tbody>
</table>
You can use this combination of determining factors in any situation where you need to identify the deliveries that are made from the UK to other EU countries.

**Define Tax Regimes**

**Features at the Tax Regime Level: Critical Choices**

Streamline your implementation by selecting the features that are applicable to the tax regime in scope. Features are used in rendering the task lists and tasks in the context of the features applicable to the tax regime in scope.

**Features**

The following table displays each feature and the impact of not selecting that feature.

---

**Warning**

Once you select a feature for a tax regime, you cannot disable it. You can enable the feature later if you do not enable it initially for a tax regime.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Impact of Not Selecting Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple Tax Jurisdictions</td>
<td>Create tax jurisdictions for a particular tax in more than one geographic region.</td>
<td>The <strong>Allow multiple jurisdictions</strong> option is not available to taxes within this tax regime.</td>
</tr>
<tr>
<td>Offset Taxes</td>
<td>Create offset taxes for tax calculation and recording of third party payables tax liabilities for reverse charges, self-assessments, and in the United States, Consumer’s Use tax.</td>
<td>The <strong>Set as offset tax</strong> option is not available to taxes within this tax regime.</td>
</tr>
<tr>
<td>Tax Exemptions</td>
<td>Create tax exemptions to apply to a specific customer or to a combination of customer and specific product.</td>
<td>The <strong>Allow tax exemptions</strong> option is not available to taxes within this tax regime.</td>
</tr>
<tr>
<td>Tax Rate Exceptions</td>
<td>Create tax exceptions to apply a special tax rate to products.</td>
<td>The <strong>Allow tax exceptions</strong> option is not available to taxes within this tax regime.</td>
</tr>
<tr>
<td>Tax Recovery</td>
<td>Create tax recovery rates for full or partial recovery of taxes paid on purchases.</td>
<td>The <strong>Allow tax recovery</strong> option is not available to taxes within this tax regime.</td>
</tr>
<tr>
<td>Tax Registration Statuses</td>
<td>Manage tax registration statuses to be used as determining factors in tax rules.</td>
<td>The <strong>Tax Registration Status</strong> field is not available for party tax profiles. You cannot use the tax registration status of <strong>Agent, Registered, or Not Registered</strong> in tax rules.</td>
</tr>
<tr>
<td>Party Fiscal Classifications</td>
<td>Manage tax classifications used by a tax authority to categorize a party and which are applicable in the tax determination process.</td>
<td>The <strong>Classifications</strong> tab is not available for party tax profiles. You cannot use party fiscal classifications in tax rules.</td>
</tr>
<tr>
<td>Fiscal Classifications</td>
<td>Description</td>
<td>Notes</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>Legal Fiscal Classifications</td>
<td>Manage classifications associated with a legal entity that represents its legal status within a country and which also guide the tax determination process.</td>
<td>The Legal Classification pages and Associated Legal Classifications region are not available for party tax profiles. You cannot use legal classifications in tax rules.</td>
</tr>
<tr>
<td>Product Category Classifications</td>
<td>Manage tax classifications for a noninventory-based product category that is used for tax determination or tax reporting purposes.</td>
<td>The Manage Product Category Fiscal Classification Codes page is not available. You cannot use product category classification codes in tax rules.</td>
</tr>
<tr>
<td>Product Fiscal Classifications</td>
<td>Manage tax classifications used by a tax authority to categorize a product for a tax and which are applicable in the tax determination process.</td>
<td>The Product Fiscal Classification pages are not available. You cannot use product fiscal classifications in tax rules.</td>
</tr>
<tr>
<td>Transaction Business Categories</td>
<td>Manage tax classifications to identify and categorize an external transaction into an Oracle Fusion Tax transaction and which are applicable in the tax determination process.</td>
<td>The Manage Transaction Business Category Codes page is not available. You cannot use transaction business category codes in tax rules.</td>
</tr>
<tr>
<td>Transaction Fiscal Classifications</td>
<td>Manage tax classifications used by a tax authority to categorize a transaction for a tax and which are applicable in the tax determination and tax reporting processes.</td>
<td>The Transaction Fiscal Classification pages are not available. You cannot use transaction fiscal classifications in tax rules.</td>
</tr>
<tr>
<td>Document Fiscal Classifications</td>
<td>Manage tax classifications used by a tax authority to categorize a document associated with a transaction for a tax and which are applicable in the tax determination and tax reporting processes.</td>
<td>The Manage Document Fiscal Classification Codes page is not available. You cannot use document fiscal classification codes in tax rules.</td>
</tr>
<tr>
<td>Intended Use Fiscal Classifications</td>
<td>Manage tax classifications based on the purpose for which a product is used and which are applicable in the tax determination process.</td>
<td>The Intended Use Fiscal Classification pages are not available. You cannot use intended use fiscal classifications in tax rules.</td>
</tr>
<tr>
<td>User-Defined Fiscal Classifications</td>
<td>Manage tax classifications for any tax requirement that you cannot define using the existing fiscal classification types.</td>
<td>The User-Defined Fiscal Classification pages are not available. You cannot use user-defined fiscal classifications in tax rules.</td>
</tr>
</tbody>
</table>

**Regimes to Rates: Explained**

Regime to rate setup contains the details of a tax regime, including all taxes, tax jurisdictions, tax statuses, and tax rates. You can update existing records or create new records at any point in the tax regime hierarchy.

Regime to rate setup tasks include:

- Tax regimes
- Taxes
- Tax jurisdictions
- Tax statuses
- Tax rates

**Tax Regimes**

Set up tax regimes in each country and geographical region where you do business and where a separate tax applies. A tax regime associates a common set of default information, regulations, fiscal classifications, and optionally, registrations, to one or more taxes. For example, in the United States create a Sales and Use Tax tax regime to group taxes levied at the state, county, and district levels.

The tax regime provides these functions:

- Groups similar taxes together
- Designates the geography within which taxes apply
- Applies as defaults the settings and values that you define for each tax in the tax regime
- Defines for which taxes the configuration options apply and a specific subscription option applies
- Provides a single registration for all taxes associated with the tax regime
- Defines the use of fiscal classifications as follows:
  - Transaction fiscal classifications
  - Product fiscal classifications
  - Party fiscal classifications

The common tax regime setup is one tax regime per country per tax type, with the tax requirements administered by a government tax authority for the entire country. There are also cases where tax regimes are defined for standard geographical types or subdivisions within a country, such as a state, province, country, or city. In these cases, you base the tax regime on the Oracle Fusion Trading Community Model standard geography.

There are more rare cases where a tax regime is based on disparate parts of a country or more than one country. In these cases, you can create one or more tax zones and set up tax regimes for these tax zones. You can also set up a tax regime as a parent tax regime to group related tax regimes together for reporting purposes.

You must set up a tax regime before you set up the taxes in the tax regime. Some tax regime values appear as defaults on the taxes that belong to the tax regime in order to help minimize tax setup.

You must associate a tax regime with all of the first party legal entities and business units that are subject to the tax regulations of the tax regime. You can set up tax configuration options when you create or edit a tax regime or when you create or edit a first party legal entity tax profile. Both setup flows appear and maintain the same party and tax regime configuration options.
Taxes

Set up details for the taxes of a tax regime. Each separate tax in a tax regime includes records for the tax statuses, tax rates, and tax rules that are used to calculate and report on the tax. Oracle Fusion Tax applies as defaults tax information from the tax regime to each tax that you create under a tax regime. You can modify this information at the tax level according to your needs, as well as add additional defaults and overrides. For tax rule defaults, specify values that apply to the majority of your transactions. Use tax rules to configure exceptions to the tax rule defaults.

Identify what taxes you must define. Each tax appears as a single tax line on a transaction. If you need to show or report more than one tax line per transaction line on a transaction, then you should set up more than one tax. For example, for US Sales and Use Tax you would define a tax for each state, county, and city.

You can create a new tax, or create a tax that is based on an existing tax within the tax regime. You do this to minimize setup by sharing tax jurisdictions and tax registrations. When you create a new tax based on an existing tax, the attributes that remain constant for all taxes derived from the source tax are not available for update. Attributes that are copied and are display only include:

- Tax regime
- Tax
- Geography information
- Tax jurisdiction settings

Note

The enable tax settings are not selected, in the same way that they are not selected when you access the Create Tax page.

You can enable a tax for simulation or for transactions only after you have completed all of the required setup.

Tax Jurisdictions

Set up tax jurisdictions for geographic regions or tax zones where a specific tax authority levies a tax. A tax jurisdiction specifies the association between a tax and a geographic location. At transaction time, Oracle Fusion Tax derives the jurisdiction or jurisdictions that apply to a transaction line based on the place of supply. You must set up at least one tax jurisdiction for a tax before you can make the tax available on transactions.

You also use tax jurisdictions to define jurisdiction-based tax rates. A tax jurisdiction tax rate is a rate that is distinct to a specific geographic region or tax zone for a specific tax. You can also create multiple jurisdictions at once using the mass create functionality for taxes that relate to specific Trading Community Model geographic hierarchies. For example, create a county jurisdiction for every county in the parent geography type of State and in the parent geography name of California.

The tax within a tax jurisdiction can have different rates for the parent and child geographies. For example, a city sales tax rate can override a county rate for the same tax. In this case, you can set up an override geography type for the city
and apply a precedence level to the city and county tax jurisdictions to indicate which tax jurisdiction takes precedence.

In addition, in some cities a different city rate applies to the incorporated area of the city, called the inner city. In these cases, you can set up an inner city tax jurisdiction with its own tax rate for the applicable customers and receivables tax. Inner city tax jurisdictions are often based on postal code groupings.

**Tax Statuses**

Set up the tax statuses that you need for each tax that you create for a combination of tax regime, tax, and configuration owner. A tax status is the taxable nature of a product in the context of a transaction and specific tax on the transaction. You define a tax status to group one or more tax rates that are the same or similar in nature.

For example, one tax can have separate tax statuses for standard, zero, exemptions, and reduced rates. A zero rate tax status may have multiple zero rates associated with it, such as Intra-EU, zero-rated products, or zero-rated exports.

You define a tax status under a tax and a configuration owner, and define all applicable tax rates and their effective periods under the tax status. The tax status controls the defaulting of values to its tax rates.

**Tax Rates**

Set up tax rates for your tax statuses and tax jurisdictions. For tax statuses, set up a tax rate record for each applicable tax rate that a tax status identifies. For tax jurisdictions, set up tax rate records to identify the tax rate variations for a specific tax within different tax jurisdictions. For example, a city sales tax for a state or province may contain separate city tax jurisdictions, each with a specific tax rate for the same tax.

You can also define tax recovery rates to claim full or partial recovery of taxes paid.

You can define tax jurisdiction and tax status rates as a percentage or as a value per unit of measure. For example, a city may charge sales tax at a rate of 8 percent on most goods, but may levy a duty tax with a special rate of 0.55 USD per US gallon on fuel. Values per unit of measure are in the tax currency defined for the tax.

You define tax rate codes and rate detail information per rate period. Rate periods account for changes in tax rates over time. A tax rate code can also identify a corresponding General Ledger taxable journal entry.

**Tax Recovery Rates**

Set up tax recovery rate codes for the recovery types identified on the taxes within a tax regime. A tax recovery rate code identifies the percentage of recovery designated by the tax authority for a specific transaction. In Canada, where more than one type of recovery is possible for a given tax, you must set up the applicable tax recovery rate codes for both the primary and secondary recovery types that can apply to a transaction.

If you set the Allow tax recovery option for a tax within a tax regime, then you must set up at least one recovery rate for the tax in order to make the tax...
available on transactions. If the recovery rate can vary based on one or more
factors, including the parties, locations, product or product purpose, then
set up tax rules to determine the appropriate recovery rate to use on specific
transactions. At transaction time, Oracle Fusion Tax uses the recovery rate
derived from the recovery tax rules, or uses instead the default recovery rate that
you define, if no recovery rate rules are defined or if no existing recovery rate
rule applies to the transaction.

**Minimum Tax Configuration: Explained**

Oracle Fusion Tax provides you with a single interface for defining and
maintaining the taxes that are applicable in each country where you do business.

The minimum tax configuration path to meet the basic tax requirements of
transactions in a given regime is a 2-step configuration process:

1. Define tax regime: This step includes the tax regime definition as well as
   the subscription by the appropriate legal entity or business unit.
2. Define transaction taxes: This step includes the basic tax definition,
   controls and defaults, direct and indirect tax rule defaults, and tax
   accounts.

The following prerequisite setups must be completed for minimum tax
configuration:

- First parties, such as legal entities and business units
- Tax geographies and zones
- Ledger and accounts
- Currency codes and exchange rates

A legal entity tax profile is automatically created when a legal entity is defined in
the implementation. Similarly, a business unit tax profile is automatically created
when a business unit is defined. For the business unit, you need to indicate
whether it will use the subscription of the legal entity instead of creating its own.

In addition, there are predefined event class mappings that describe the mapping
between an application event class and the corresponding tax event class. For
example, the tax determination process for a sales debit memo and sales invoice
are essentially the same. These two application event classes correspond to the
same tax event class namely, a sales transaction. Although you cannot update
the event class mappings, you can set up configuration specific event class
mappings.

**Define Tax Regime**

The first step includes the tax regime definition and subscription by an
appropriate legal entity or business unit. While creating your tax regime, you
can minimize configuration and maintenance costs by creating content that can
be shared by more than one entity. For example, legal entities can subscribe
to the shared reference data instead of creating separate and repetitive data.
If the subscribing legal entities have some variations in their setup, you can
create override data to meet the specific exceptions that are applicable to these
organizations.
Use Oracle Fusion Tax features to enable only those features that are relevant to taxes in the tax regime. Based on the features you select, the subsequent setup pages and task lists for the tax regime are rendered or hidden.

Define Transaction Taxes

The second step includes basic tax definition, such as geographic information, controls and defaults, direct and indirect tax rule defaults, and tax accounts.

The basic tax definition includes controls that you can set to provide the override capability at transaction time. For example, if you want to allow users to make manual updates on transaction tax lines, select the **Allow override for calculated tax lines** and the **Allow entry of manual tax lines** options. However, if you want to enforce automatic tax calculation on transaction tax lines, do not enable these options.

Use the direct and indirect tax rule defaults to specify the values that apply to the majority of your transactions. Create tax rules to address the exceptions or variations to the defaults. For example, for the Goods and Services Tax (GST) that applies to the supply of most goods and services in Canada, set the Tax Applicability direct tax rule default to **Applicable**. A luxury tax, on the other hand, is a tax on luxury goods or products not considered essential. As it would not apply to most goods and services, set the Tax Applicability direct tax rule default to **Not Applicable**, and create a tax rule to make the tax applicable when the product in the transaction satisfies the luxury requirement.

Assign your default tax accounts for the taxes in a tax regime to post the tax amounts derived from your transactions. The tax accounts you associate serve as default accounting information for taxes, tax rates, tax jurisdictions, and tax recovery rates. The tax accounts you define at the tax level, default to either the tax rate accounts or tax jurisdiction accounts for the same tax and operating unit, depending upon the tax accounts precedence level of the tax regime. You can update these default tax accounts in the tax rate or tax jurisdiction setup.

Minimum Tax Configuration: Points to Consider

The minimum tax configuration setup must be designed to handle the majority of tax requirements. As part of defining transaction taxes, decide the direct and indirect tax rule defaults for the tax and set up the associated tax accounts.

For complex tax requirements, create tax rules that consider each tax requirement related to a transaction before making the final tax calculation. During the execution of the tax determination process, Oracle Fusion Tax evaluates, in order of priority, the tax rules that are defined against the foundation tax configuration setup and the details on the transactions. If the first rule is successfully evaluated, the result associated with the rule is used. If not, the next rule is evaluated until either a successful evaluation or default value is found.

Setting Up Direct Tax Rule Defaults

The direct tax rule defaults are the default values for the direct tax rule types, which include:

- Place of supply
• Tax applicability
• Tax registration
• Tax calculation formula
• Taxable basis formula

Place of Supply

Use the Place of Supply direct tax rule default to indicate the specific tax jurisdiction where the supply of goods or services is deemed to have taken place. For example, in Canada, the place of supply for GST is typically the ship-to location. To handle the majority of Goods and Services Tax (GST) transactions, select Ship to as your default place of supply.

Note

The corresponding place of supply differs based on the type of transaction. For example, a place of supply of Ship to corresponds to the location of your first party legal entity for Payables transactions. For Receivables transactions, Ship to corresponds to the location of your customer site. For exceptions to this default, create Determine Place of Supply rules.

Tax Applicability

Use the Tax Applicability direct tax rule default to indicate whether the tax is typically applicable or not applicable on transactions. For example, the GST in Canada is a tax that applies to the supply of most property and services in Canada. When you create the GST tax, select Applicable as your default tax applicability. For exceptions to this default, create Determine Tax Applicability rules.

Tax Registration

Use the Tax Registration direct tax rule default to determine the party whose tax registration status is considered for an applicable tax on the transaction. For example, with a direct default of bill-to party, Oracle Fusion Tax considers the tax registration of the bill-to party and stamps their tax registration number onto the transaction, along with the tax registration number of the first party legal reporting unit. For exceptions to this default, create Determine Tax Registration rules.

Tax Calculation Formula

Use the Tax Calculation Formula direct tax rule default to select the formula that represents the typical calculation of tax for a transaction line. A common formula, STANDARD_TC, is predefined, where the tax amount is equal to the tax rate multiplied by the taxable basis. For exceptions to this default, create Calculate Tax Amounts rules.

Taxable Basis Formula

Use the Taxable Basis Formula direct tax rule default to select the formula that represents the amount on which the tax rate is applied. The following common formulas are predefined:
- **STANDARD_TB**: The taxable basis is equal to the line amount of the transaction line.
- **STANDARD_QUANTITY**: The taxable basis is equal to the quantity of the transaction line.
- **STANDARD_TB_DISCOUNT**: The taxable basis is the line amount of the transaction line less the cash discount.

For exceptions to this default, create Determine Taxable Basis rules.

### Setting Up Indirect Tax Rule Defaults

The indirect tax rule defaults for a tax include:

- Tax jurisdiction
- Tax status
- Tax recovery rate
- Tax rate

#### Tax Jurisdiction

Use the Tax Jurisdiction indirect tax rule default to indicate the most common geographic area where a tax is levied by a specific tax authority. For example, value-added tax (VAT) is applicable to the supply of most goods and services in Portugal. For the tax PT VAT, create the default tax jurisdiction as the country of Portugal. To address specific tax regions such as Azores and Madeira, which have lower VAT rates than Portugal, define jurisdiction rates with different VAT rates.

#### Tax Status

Use the Tax Status indirect tax rule default to indicate the taxable nature of the majority of your transactions. For example, if your operations primarily include zero-rated transactions, select the default tax status as **Zero** instead of **Standard**. This setting facilitates tax determination when multiple zero rates are defined to handle different reporting requirements for zero rate usage, such as intra-EU, zero-rated products, or zero-rated exports. For exceptions to this default, create Determine Tax Status rules.

#### Tax Recovery

Use the Tax Recovery rate indirect tax rule default to indicate the recovery rate to apply to each recovery type for each applicable tax on a purchase transaction. For example, in Canada, both federal and provincial components of Harmonized Sales Tax (HST) are 100% recoverable on goods bought for resale. In this case, with two recovery types, you can set up two recovery rate defaults for the HST tax. For exceptions to this default, such as when the recovery rate determination is based on one or more transaction factors, create Determine Recovery Rate rules.

#### Tax Rate

Use the Tax Rate indirect tax rule default to specify the default tax rate that is applicable to the majority of your transactions associated with this tax. You can
create additional tax setup, such as jurisdiction rates, or create tax rules to set alternate values as required. For example, HST in Canada is applied at a 13% rate in most provinces that have adopted HST, except for British Columbia where the rate is 12% and Nova Scotia where the rate is 15%. To satisfy this requirement a single rate of 13% can be defined with no jurisdiction and then a 12% rate can be defined and associated with the British Columbia jurisdiction (15% rate assigned to Nova Scotia). This minimizes the setup required by creating an exception based setup. For exceptions to this default, create Determine Tax Rate rules.

**Setting Up Tax Accounts**

Set up tax accounts at the tax level. The application automatically copies the tax account combination to the tax rates that you subsequently create for the tax for the same ledger and optionally, the same business unit.

Define tax accounts at any of the following levels. The defaulting option is only available at the tax level.

- Tax
- Tax jurisdiction
- Tax rate
- Tax recovery rate

**Note**

This is a one-time defaulting opportunity. Any subsequent changes at the account level are not copied to the tax rate level nor are they used during the AutoAccounting process. Changes at the tax level do impact tax account defaulting when you create new tax rates.

Setting up tax accounts comprise of specifying the following:

- **Ledger and Business Unit**: The ledger and business unit for which you are creating the tax accounts.

- **Interim Tax**: An account that records tax recovery or liability until the event prescribed by the statute is complete. Generally, the payment of the invoice is the event that triggers the generation of the tax recovery or liability. You must set up an interim tax account for taxes and tax rates that have a deferred recovery settlement. Once you set up an interim tax account for this tax rate, you cannot change the recovery settlement to **Immediate**.

- **Tax Recoverable or Liability Account**: An account that records tax recovery amounts or relieves tax liability amounts. If you set up recovery rates for a tax that you also intend to self-assess, then define a tax recovery account for the associated recovery rates and a tax liability account for the associated tax rates.

- **Finance Charge Tax Liability**: An account that records the tax liability associated with finance charges that is used as a deduction against overall tax liability.
• **Nonrecoverable Tax Accounts**: Accounts that record tax amounts on earned and unearned discounts and adjustments that you cannot claim as a deduction against tax liability.

• **Expense and Revenue Accounts**: Accounts that record net changes generated by adjustments, earned and unearned discounts, and finance charges. Receivables activities such as discounts and adjustments reduce the receivable amount, and are therefore considered an expense.

**Minimum Tax Configuration: Worked Example**

The following example illustrates the minimum tax configuration setup to meet the basic requirements in Canada for the Goods and Services Tax (GST). You set up a tax regime for both GST and Harmonized Sales Tax (HST). One recovery type is created for the fully recoverable status of the transaction.

In Canada, GST is a tax that applies to the supply of most property and services in Canada. The provinces of British Columbia, Ontario, New Brunswick, Nova Scotia, and Newfoundland and Labrador, referred to as the participating provinces, combine their provincial sales tax with GST to create HST. Generally, HST applies to the same base of property and services as the GST. Every province in Canada except Alberta has implemented either provincial sales tax or the HST. In countries like Canada, some or all taxes on business transactions for registered companies are recoverable taxes.

ABC Corporation is a business with a chain of bookstores across Canada. It intends to implement the Oracle Fusion Tax solution at its store in the province of Alberta. The GST rate of 5% is applicable for sales in Alberta. Input Tax Credit is available for GST included in purchases. ABC Corporation's primary ledger is CA Ledger, and the business unit is CA Operations. The tax account 0001-1500-1100-1000 is reserved for the **Tax Recoverable or Liability** account.

The tax implications in this scenario are:

• Five percent (5%) GST is applicable on the sale of goods in Alberta

• Neither the HST nor provincial sales tax applies in Alberta

• Place of supply for GST tax is generally based on the place of delivery or ship-to location.

To determine the GST tax in Alberta, perform the following steps:

1. Define tax regime
2. Define transaction taxes
3. Create the direct tax rule defaults
4. Create the indirect tax rule defaults
5. Enable tax

**Define Tax Regime**

1. On the Create Tax Regime page, enter the tax regime code for GST and HST in Canada.

---

**Note**
Use a coding convention to indicate both the country and the type of tax that belongs to this regime. For example, CA GST and HST.

2. Select the regime level to define the geographic area of the tax treatment. The option selected must depict the need for the tax regime. It should be set to Country for all federal taxes.

3. Specify Canada as the country for which this tax regime is being defined.

4. Enter a start date that will appear as a default to all related tax setup within the tax regime.

**Note**

Consider your tax planning carefully before entering the start date. This date must accommodate the oldest transaction that you want to process within this tax regime. After you create the tax regime, you can only update this date with an earlier date. If you enter an end date, you cannot update this date after you save the record.

5. Enter tax currency. Enter CAD, which is the three-letter ISO code for the Canadian dollar.

Tax currency is the currency required by the tax authority. Use the tax currency to pay the tax authority and to report on all tax transactions.

6. Select the Allow cross regime compounding option to set taxes within the tax regime to be based on the calculation of, or compounded on, taxes in another tax regime.

For example, in Quebec, the provincial sales tax is applied to both the selling price and GST. Enter a value as the compounding precedence to indicate the order of cross regime compounding. A lower number indicates that the tax regime will be processed first. Allowing gaps between numbers provide flexibility in the event that another higher priority tax regime is introduced in the future.

7. On the Configuration Options tab, select the party name that identifies either the legal entity or the business unit or both for which you will define the configuration options.

8. For the Configuration of Taxes and Rules, select the subscription that defines the configuration owner setup that will be used for transactions of the specific legal entity and business unit for this tax regime.

This selection also defines whether any shared content can be overridden by the subscribing party to allow unique, separate setup for certain tax content.

9. Enter the effective start date for this configuration option. Enter a date range that is within the date range of both the party tax profile and the tax regime.

**Define Transaction Taxes**

1. On the Create Tax page, enter the name of the tax regime that you created in the Define Tax Regime step, such as CA GST and HST.
2. Select the configuration owner for this tax. To minimize configuration and maintenance costs, select **Global Configuration Owner** as the configuration owner.

3. Enter the name of the tax you are defining, such as CA GST.

4. Select **Province** as the geography type.

5. To minimize setup and maintenance costs, specify the highest-level parent geography type (Country), unless the tax is only applicable to a specific geography. Select **Country** from the list of values. For the parent geography name, enter **Canada**.

6. Enter a value as the compounding precedence to reflect the order of tax compounding. A lower number indicates that a tax is processed first. Allowing gaps between numbers provide flexibility in the event that another higher priority tax is introduced in the future.

7. Enable the **Allow override of calculated tax lines** option to allow users to override the automatic tax calculation on invoice tax lines.

8. Enable the **Allow multiple jurisdictions** option to define tax jurisdictions for this tax in more than one geographic region.

9. Enable the **Allow mass create of jurisdictions** option to enable mass creation of tax jurisdictions for this tax, which allows you to create multiple jurisdictions at the same time.

10. Enable the **Allow tax recovery** option.

11. Enable the **Allow tax recovery rate override** option if you want to allow user override of the calculated tax recovery rate on transaction lines.

12. Select **Standard** as the primary recovery type.

**Assign Tax Accounts**

1. Navigate to the Tax Accounts tab.

2. Select CA Ledger as the primary ledger to use for tax accounts and CA Operations as the business unit.

3. Enter 0001-1500-1100-1000 as the Tax Recoverable or Liability account.

**Create Direct Tax Rule Defaults**

1. Navigate to the Tax Rule Defaults tab.

2. Select **Ship to** from the Place of Supply list of values, to specify the default.

3. Select **Applicable** from the Tax Applicability list of values to specify the Tax Applicability default.

4. Select **Ship-from party** to specify the Tax Registration default.

5. Select **STANDARD_TC** as the Tax Calculation Formula default.

6. Select **STANDARD_TB** as the Taxable Basis Formula default.

**Create Indirect Tax Rule Defaults**

1. Select **Tax Jurisdiction** as your rule type and create the rule type default. In the Tax Jurisdiction Code field, enter a tax jurisdiction code for the province of Alberta, such as CA Alberta. Select **Province** as the
Define Common Procurement Configuration

1. Click the **Enable tax for simulation** option. This allows you to verify the tax configuration using the Tax Simulator.

2. Once you have verified your tax configuration with simulated transactions, click the **Enable tax for transactions** option. This allows you to use this tax in transaction processing.

3. Click **Save and Close**.

   For ABC’s transactions in the province of Alberta, the following is determined by default:
   - GST tax is applicable and will be calculated at a percentage rate of 5%.
   - 100% of the GST can be recovered.

**Associated Taxes Setup for a Tax Regime: Explained**

When you create a tax regime, you specify the options and defaults available to the taxes associated with the tax regime. You also enable the features that are applicable to the tax regime and its taxes.

The options appearing in the Associated Taxes Setup Information region on the Edit Tax Regime page are a result of the features enabled and the options you selected at the tax level. These options include:

- **Allow multiple jurisdictions**
- **Allow tax recovery**
- **Allow tax exceptions**
- **Allow tax exemptions**

The preceding options always appear as read-only check boxes in the Associated Taxes Setup Information region. The option appears as selected if you selected
the option in one of the taxes within this tax regime. If you did not select the option in one of the taxes, then the option appears as not selected.

For example, suppose you have a California county sales tax that applies to all counties, so you need a tax with multiple jurisdictions. In this case, you must enable the Multiple Jurisdictions feature at the tax regime level and then select the Allow multiple jurisdictions option at the tax level. When you access the Edit Tax Regime page, Associated Taxes Setup Information region for this tax regime, the Allow multiple jurisdictions option appears as selected.

**Manage Controls and Defaults**

**Tax Regime Controls and Defaults: Points to Consider**

A tax regime associates a common set of default information, regulations, fiscal classifications, and optionally, registrations, to one or more taxes. Set up tax regimes in each country and geographical region where you do business and where a separate tax applies.

The tax regime setup details include:

- Designating the geography to which taxes within a tax regime apply
- Defining the controls and defaults that apply to taxes and associated lower level information
- Specifying configuration options and service subscriptions

**Designating the Geography**

The common tax regime setup is one tax regime per country per tax type, but you can also have tax regimes based on parts of a country or more than one country. Select the regime level as:

- **Country**: The tax regime is applicable to a specific country.
- **Tax zone**: The tax regime is applicable to parts of a country or more than one country. Enter the tax geography type and tax geography name associate with the group of countries or the tax zone that you want. The tax geography type and tax geography name correspond to the tax zone type and tax zone respectively.

If applicable, designate the tax regime as a parent regime or indicate the parent regime name if the tax regime belongs to a parent regime. Use a tax regime defined as a parent tax regime to group other nonparent tax regimes for reporting purposes.

**Defining Controls and Defaults**

Set tax-level controls to enable the options that you want to make available to the taxes in this tax regime. If necessary, you can disable the options that you enable here for individual taxes within the tax regime. Enter default values for the taxes in this tax regime. You can update the default values at the tax level.
If you disable a controlled option at the tax regime level it is not available as an option at the tax level.

The following table describes the defaults and controls available at the tax regime level.

Defaults Region

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Derived from</th>
<th>Default Appears on</th>
<th>Controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Currency</td>
<td>The default currency of the taxes within this tax regime</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Minimal Accountable Unit</td>
<td>The minimal unit of currency that is reported to the tax authority, for example, 0.05 GBP indicates that 5 pence is the minimal unit</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Tax Precision</td>
<td>A one digit whole number to indicate the decimal place for tax rounding</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Tax Inclusion Method</td>
<td>A method that describes whether the line amount includes tax or excludes tax</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Conversion Rate Type</td>
<td>The specific exchange rate table that is used to convert one currency into another, for example, the Association of British Travel Agents exchange rate used in the travel industry</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Rounding Rule</td>
<td>The rule that defines how rounding is performed on a value, for example, up to the next highest value, down to the next lower value, or to the nearest value</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Allow tax rounding override</td>
<td>Allow the override of the rounding defined on the tax registration records</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Reporting Tax Authority</td>
<td>The default tax authority to whom the tax reports are sent</td>
<td>None</td>
<td>• Tax • Tax registration</td>
<td>None</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------</td>
<td>------</td>
<td>--------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Collecting Tax Authority</td>
<td>The default tax authority to whom the tax is remitted</td>
<td>None</td>
<td>• Tax • Tax registration</td>
<td>None</td>
</tr>
<tr>
<td>Default Settlement Option</td>
<td>A lookup code to indicate whether an input tax is recovered when an invoice is recorded or only when the invoice is paid and whether an output tax is due for settlement when the invoice is issued or only when the payment is received against it</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Use legal registration number</td>
<td>Option that controls whether the tax registration number is the same as the legal registration number of the party</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
</tbody>
</table>

**General Controls Region**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Derived from</th>
<th>Default Appears on</th>
<th>Controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow override and entry of inclusive tax lines</td>
<td>Option that controls whether you can override and enter inclusive or exclusive line amounts</td>
<td>None</td>
<td>Tax</td>
<td>None</td>
</tr>
<tr>
<td>Use tax reporting configuration</td>
<td>Option that controls whether the tax reporting details are available on the first party tax registration record for this tax regime</td>
<td>None</td>
<td>None</td>
<td>Controls whether you can enter tax reporting configuration details on the tax registration for this tax regime for your first parties</td>
</tr>
</tbody>
</table>

**Compounding Level Controls Region**
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Derived from</th>
<th>Default Appears on</th>
<th>Controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow cross regime compounding</td>
<td>Option that controls whether cross regime compounding is needed for this tax regime</td>
<td>None</td>
<td>None</td>
<td>Controls whether this tax regime is compounded based on the tax calculated from another tax regime</td>
</tr>
<tr>
<td>Compounding Precedence</td>
<td>Defines the order in which taxes within the compound tax regimes need to be calculated. A tax within a tax regime with a lower value is calculated first.</td>
<td>None</td>
<td>None</td>
<td>Controls the order in which taxes within tax regimes are calculated</td>
</tr>
</tbody>
</table>

**Important**

Oracle Fusion Tax provides features at the tax regime level to streamline your implementation by selecting the features that are applicable to the tax regime in scope. You must enable the features to use that functionality for the tax regime and related taxes.

**Specifying Configuration Options and Service Subscriptions**

Set up configuration options to associate tax regimes with the parties in your company that have a tax requirement under these tax regimes. You can set up tax configuration options when you create a tax regime or when you create a party tax profile for a first party legal entity or business unit. Both tax regime and party tax profile setup flows appear and maintain the same party and tax regime association. Configuration options only apply to tax regimes directly linked to taxes and not to tax regimes that are used to group other tax regimes.

Oracle Fusion Tax lets you use the tax services of external service providers for tax calculation of US Sales and Use Tax on receivables transactions. The setup for provider services is called a service subscription. A service subscription applies to the transactions of one configuration option setup for a combination of tax regime and legal entity or business unit.

**Note**

The level of detail of tax rounding definitions for the taxes in the tax regime must equal or exceed the level of detail of the service provider tax rounding definitions.

**Inclusive Taxes: Explained**

Calculating tax on a transaction as inclusive of the line amount is generally a business decision. This decision is based on the relationship between the transacting parties and the items or taxes involved.
Taxes applicable on a transaction are made inclusive of the item line amount either:

- Manually
- Automatically

**Manual Approach**

In the manual approach, you access the calculated tax lines on a transaction and select the **Inclusive** option. This action includes the calculated tax amount with the item value.

However, this option is controlled through two factors:

- Privileges are assigned to the users for accessing and editing the calculated tax lines.
- Setup restrictions are applied to edit the **Inclusive** option on the calculated tax lines.

**Automatic Approach**

In the automatic approach, you can configure the tax setup and calculate the tax on a transaction as inclusive of the item line amount. Since this requirement is primarily driven by the tax legislation and the business relationship between the transacting parties, the option for configuring the inclusiveness is made available on the tax and tax rate definition and the third party and legal reporting unit tax profiles on the tax registration and general data tabs. The tax determination process uses a hierarchy approach to evaluate the defined setup and applies the inclusiveness option on the transaction.

In tax setup there are options to choose for applying the inclusiveness on a transaction. They are:

- **Standard noninclusive handling**: This option calculates the taxes as exclusive of the given transaction line amount.
- **Standard inclusive handling**: This option calculates the taxes as inclusive of the given transaction line amount.
- **Special inclusive handling**: This option calculates the taxes as inclusive of the given transaction line amount, but the calculation methodology differs from the standard inclusive process.

The following table illustrates the calculation methodology used with each of these options when a transaction line amount is 1000 USD and the applicable tax rate is 10% of the taxable basis amount, for example, line amount:

<table>
<thead>
<tr>
<th>Method</th>
<th>Calculation</th>
<th>Taxable Basis Amount</th>
<th>Tax Amount</th>
<th>Transaction Line Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Noninclusive</td>
<td>1000 USD * 10/100</td>
<td>1000 USD</td>
<td>100 USD</td>
<td>1100 USD</td>
</tr>
<tr>
<td>Standard Inclusive</td>
<td>1000 USD * 10/110</td>
<td>909.09 USD</td>
<td>90.91 USD</td>
<td>1000 USD</td>
</tr>
<tr>
<td>Special Inclusive</td>
<td>1000 USD * 10/100</td>
<td>900 USD</td>
<td>100 USD</td>
<td>1000 USD</td>
</tr>
</tbody>
</table>
Tax Amount Rounding: Explained

Taxes applicable on a transaction are generally calculated as the taxable basis multiplied by the tax rate equals the tax amount. This calculated amount can result in an odd value or with a large number of decimal place. You can configure the tax setup to adjust or round the tax calculation according to the specific requirements of the transacting parties and tax authority or to the accepted currency denominations.

Key parameters that influence the rounding of calculated tax amount are:

- Tax precision: The number of decimal places to which to calculate the tax amount.
- Minimum accountable unit: The smallest currency unit that a tax amount can have.
- Rounding level: The transaction level at which the rounding is to be performed. The available options are Header and Line.
- Rounding rule: The method that is used to round off the calculated taxes to the minimum accountable unit. The available options are Up, Down, and Nearest.

Define the key parameters at various places within Oracle Fusion Tax. The rounding process derives the tax precision and minimum accountable unit details from the tax setup. The rounding process derives the rounding rule and rounding level details through the predefined processing hierarchy involving:

- Configuration owner tax options defined for the configuration owner and event class
- Event class options for the event class
- Party tax profiles of the parties or party sites as given in the rounding precedence of the configuration owner tax options or in the derived registration party
- Tax

Note

If you plan to use a third party service provider then you must define tax rounding information that is at least as detailed as the rounding information of the service provider.

Manage Configuration Options and Service Subscriptions

Configuration Options: Explained

Set up configuration options to associate tax regimes with the parties in your company that have a tax requirement under these tax regimes.

There are two fundamentally different approaches to tax configuration options namely:
• Using tax configuration setup defined within Oracle Fusion Tax.
• Using an external tax service provider.

**Using Tax Configuration Setup Defined Within Oracle Fusion Tax**

Use the tax configuration setup in Oracle Fusion Tax to calculate, record, and account for transaction taxes on transaction taxable transactions.

The following concepts control how this setup is managed, used, and shared:

• Tax configuration owner
• Tax content subscription
• Existing tax option

**Tax Configuration Owner**

The tax configuration owner is a business unit, legal entity, or the global configuration owner that owns the data. The global configuration owner is an abstract owner which is used to define the owner of content that can be shared by any business units and first party legal entities.

Identify a specific first party legal entity as a parent first party organization to allow the configuration to be owned by a specific first party and shared by other parties. You can then share this setup with another first party legal entity or business unit for their transactions. Use a parent first party organization tax configuration to share among a group of first party organizations but you still have the tax setup managed by a single first party organization.

In the case of global configuration owner, if you are assigned the Create Tax Regime privilege, you have update rights to all tax configuration data maintained by the global configuration owner.

**Tax Content Subscription**

Use tax content subscriptions to define which configuration owner’s setup is used for transactions for a specific first party legal entity or business unit for a specific tax regime. Also, use tax content subscriptions to specify whether any shared content can be overridden by the subscribing party to allow unique, separate setup for certain tax content.

Party override is permitted for the following setup:

• Tax
• Tax status
• Tax rate
• Tax recovery rate
• Tax rules
  
  Do this indirectly by adding higher priority rules specific to the subscribing first party legal entity or business unit.

The content subscription options are:
Define Common Procurement Configuration

### Tax Content Subscription

<table>
<thead>
<tr>
<th>Tax Content Subscription</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common configuration</td>
<td>For tax processing, the tax determination process uses the shared tax content defined and maintained by the global configuration owner.</td>
</tr>
<tr>
<td>Party-specific configuration</td>
<td>The specified first party organization defines and maintains its own tax content. For tax processing, the tax determination process uses only the tax content owned by the specific first party legal entity or business unit.</td>
</tr>
<tr>
<td>Common configuration with party overrides</td>
<td>This option is similar to the common configuration in that it allows you to use tax content owned by the global configuration owner. However, you can also maintain party-specific content which is used in preference to the common configuration content. In the absence of tax content owned by the specific first party organization, the tax determination process uses the tax content owned by the global configuration owner.</td>
</tr>
<tr>
<td>Parent first party organization with party overrides</td>
<td>This option is similar to the common configuration with party override subscription except instead of the tax content being owned by the global configuration owner it is owned by a specific first party legal entity. You can override the specific first party setup.</td>
</tr>
</tbody>
</table>

A similar concept is used to define where you use tax exceptions for a specific tax configuration. The tax subscription option available for product exceptions is dictated to some extent by the main tax content subscription as follows:

<table>
<thead>
<tr>
<th>Options Defined for Tax Content Subscription</th>
<th>Content Subscription Options Available for Product Exceptions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common configuration</td>
<td>Common configuration</td>
<td>For tax processing, the tax determination process uses tax exceptions defined and maintained by the global configuration owner.</td>
</tr>
<tr>
<td>Party-specific configuration</td>
<td>Party-specific configuration</td>
<td>The specified first party organization defines and maintains its own tax exceptions. For tax processing, the tax determination process uses only the tax exceptions owned by the specific first party organization.</td>
</tr>
<tr>
<td>Common configuration with party overrides</td>
<td>Common configuration</td>
<td>For tax processing, the tax determination process uses tax exceptions defined and maintained by the global configuration owner.</td>
</tr>
<tr>
<td>Common configuration with party overrides</td>
<td>Party-specific configuration</td>
<td>The specified first party organization defines and maintains its own tax exceptions. For tax processing, the tax determination process uses only the tax exceptions owned by the specific first party organization.</td>
</tr>
</tbody>
</table>
Set up tax configuration options when you create a tax regime or when you create a party tax profile for a first party legal entity or business unit. Both setup flows display and maintain the same party or regime definitions. Specify effective start and end dates to identify which configuration should be used based on the transaction date. You can enable the business unit so that Oracle Fusion Tax automatically uses the configuration of the legal entity. Once you set this option the application records the date it occurred as the start date. This date is used and compared to the transaction dates to identify if the application uses the legal entity subscription in preference to the subscription of the business unit. The specific first party legal entity that is used is defined by the legal entity associated with the transaction.

**Existing Tax Option**

Copy a tax from an existing tax in the Manage Taxes page to share tax registrations and tax jurisdictions while maintaining two versions of the same tax, owned by two different tax configuration owners each with their own tax statuses, tax rates, and tax rules. For example, this is useful when you set up US sales and use tax that requires a significant number of tax registrations and tax jurisdictions.

**Using External Tax Service Provider**

Oracle Fusion Tax lets you use the tax services of external service providers for tax calculation of US Sales and Use Tax on Receivables transactions. Oracle Fusion Tax provides transparent integration between the external provide tax service and Oracle Fusion Receivables.

You can use the tax services of these external service providers:

- Taxware, LP: a First Data Company
- Vertex, Inc.

The setup for provider services is called a service subscription. A service subscription applies to the transactions of one configuration option setup for a combination of tax regime and legal entity or business unit. Set up service subscriptions when you create a tax regime or when you create a party tax profile for a first party legal entity or business unit. Specify effective start and end dates to identify which configuration should be used based on the transaction date.

**Content Subscriptions: Critical Choices**

Choose which of the following tax content subscription options to use to optimize your tax setup:

- Whether to use service subscriptions versus Oracle Fusion tax content.
- What type of tax configuration options to use.
- When to change from business unit to using tax configuration at the first party legal entity.
• When to use create from an existing tax option.

**Using a Service Subscription Versus Oracle Fusion Tax Content**

Use the tax services of external service providers where tax content is required for Receivables transactions for a significant number of tax jurisdictions. You should not use a service provider if their use is not needed to support US Sales and Use Tax regimes or you need to create and maintain tax regimes outside of the United States.

You can use the tax services of these external service providers:

- Taxware, LP: a First Data Company
- Vertex, Inc.

**Using Tax Configuration Options**

If you decide not to use an external service provider or you need to create tax content for tax regimes outside the US then create and maintain your tax content in Oracle Fusion Tax.

Once the decision is made to use Oracle Fusion Tax you need to choose the level of tax configuration options. Sharing tax content prevents the need for duplicate maintenance with its inefficiencies and potential inconsistencies. Consider these scenarios and options:

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>You have a single central corporate tax center responsible for maintenance of tax setup for all legal entities and business units.</td>
<td>Use the common configuration with party override option. This allows a single tax setup to be created and maintained by the corporate tax center.</td>
</tr>
<tr>
<td>You need to have strict control of who can maintain the tax content.</td>
<td>Use the common configuration option. By not allowing party override you restrict the access to the global configuration owner to an authorized user who can maintain all of the tax content.</td>
</tr>
<tr>
<td>You have regional centers responsible for tax content.</td>
<td>Use the parent first party configuration with party override option. This permits a regional setup with an actual or logical parent legal entity to be created and maintained by each regional center.</td>
</tr>
</tbody>
</table>

Even if there is no obvious need to share tax configuration, for example, there is only a single first party legal entity operating in each tax regime, significant business events such as takeovers or mergers may mean that there could be a future need to share content. In this case the original first party legal entity can act as the configuration owner and then any subsequent first party can subscribe to the first party’s content using the parent first party configuration with party override. Alternatively, set up the original tax content using global configuration owner in preparation for any future business event that requires tax content to be shared.

**Changing from Business Unit to Using Tax Configuration at the First Party Legal Entity**

If you can standardize your tax setup across all business units for a given legal entity then consider moving to configuring and using tax setup at the legal entity.
level. Set the **Use subscription of the legal entity** option on the business unit tax profile. Oracle Fusion Tax records the date this occurs and compares it to the transaction date to identify if the legal entity subscription should be used in preference to the subscription to the business unit.

**Using Create from an Existing Tax Option**

Create a tax from an existing tax when you have a need to share tax jurisdictions and tax registrations. You maintain the tax jurisdictions and tax registrations once for taxes with the same name within the same tax regime owned by different configuration owners.

**Tax Configuration Options in the Tax Determination Process: How They Are Used**

At transaction time the owner of the transaction derives the configuration options that are used. When you enter a transaction for a given first party organization, the tax data applied to that transaction is determined by the configurations defined for the combination of that first party organization (business unit or first party legal entity) and the tax regime derived from the addresses or from the tax classification codes used on the transaction.

**Settings That Affect the Application of Tax Data on Transactions**

Use tax content subscriptions to define which configuration owner’s setup is used for transactions for a specific first party legal entity or business unit for a specific tax regime. Also, use tax content subscriptions to specify whether any shared content can be overridden by the subscribing party to allow unique, separate setup for certain tax content.

Tax content subscription options are:

- Common configuration
- Party-specific configuration
- Common configuration with party overrides
- Parent first party organization with party overrides

**How Tax Data Is Determined**

Based on the defaults and tax rules you have defined, tax data is applied to transactions as follows:

<table>
<thead>
<tr>
<th>Configuration for Taxes and Rules Option</th>
<th>Tax Content Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common configuration</td>
<td>• The tax determination process uses only the tax content owned by the global configuration owner.</td>
</tr>
<tr>
<td></td>
<td>• If you manually override tax information on the transaction only tax content owned by the global configuration owner is displayed in the list of valid values available.</td>
</tr>
</tbody>
</table>
Party-specific configuration

- The tax determination process uses only the tax content owned by the first party organization, business unit or first party legal entity, for whom the transaction is being entered.

- If you manually override tax information on the transaction only tax content owned by the first party organization is displayed in the list of valid values available.

Note

For the first party organization it can be the business unit owning the tax content or the first party legal entity-owned setup depending on the specific subscription being used.

Common configuration with party overrides

- The tax determination process uses any tax content owned by the first party for whom the transaction is being entered. In the absence of tax content owned by that first party organization, the tax determination process uses tax content owned by the global configuration owner.

- If you manually override tax information on the transaction both the override tax content owned by the specific first party and the tax content owned by the global configuration owner that you have not overridden are displayed in the list of valid values available.

Parent first party organization with party overrides

- The tax determination process uses any tax content owned by the first party for whom the transaction is being entered. In the absence of tax content owned by the first party organization, the tax determination process uses tax content owned by the parent first party organization.

- If you manually override tax information on the transaction both the override tax content owned by the specific first party and the tax content owned by the designated parent first party organization that you have not overridden are displayed in the list of valid values available.

If you are using product exceptions, those exceptions are applied to the transactions as shown in the following table:

<table>
<thead>
<tr>
<th>Configuration for Product Exceptions</th>
<th>Tax Exceptions Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common configuration</td>
<td>The tax determination process uses only the tax exceptions defined and maintained by the global configuration owner.</td>
</tr>
<tr>
<td>Party-specific configuration</td>
<td>The tax determination process uses only the tax exceptions owned by the specific first party organization.</td>
</tr>
</tbody>
</table>
Setting Up Tax Configuration Options: Worked Example

This example demonstrates how you set up the appropriate tax configuration options for your company that has three regional centers. These centers are responsible for tax setup and maintenance among other corporate activities. Each of these regional corporate centers is associated with a first party legal entity and business unit.

Your company has their regional centers in:

- North America (NAM), based in Redwood City, California, US
- Asian and Pacific (APAC), based in Melbourne, Australia
- Europe, Middle East, and Africa (EMEA), based in London, UK

Each country has a single first party legal entity with a single business unit, except for:

- Countries which have the regional corporate centers have a first party legal entity and business unit for each corporate center.
- Sales, marketing, and manufacturing organization has a first party legal entity and business unit.

Create tax regimes for each country and the appropriate tax configuration options.

Prerequisites

To create the appropriate tax configurations, you must set up the following:

1. The legal entities for:

<table>
<thead>
<tr>
<th>First Party Legal Entity</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMEA LE</td>
<td>UK</td>
</tr>
<tr>
<td>GB LE</td>
<td>GB</td>
</tr>
<tr>
<td>FR LE</td>
<td>FR</td>
</tr>
<tr>
<td>DE LE</td>
<td>DE</td>
</tr>
<tr>
<td>APAC LE</td>
<td>AU</td>
</tr>
<tr>
<td>AU LE</td>
<td>AU</td>
</tr>
<tr>
<td>SI LE</td>
<td>SI</td>
</tr>
<tr>
<td>NZ LE</td>
<td>NZ</td>
</tr>
<tr>
<td>NAM LE</td>
<td>NAM</td>
</tr>
<tr>
<td>US LE</td>
<td>US</td>
</tr>
<tr>
<td>CA LE</td>
<td>CA</td>
</tr>
</tbody>
</table>

2. The sales, marketing, and manufacturing organization's business unit uses the tax configuration of the legal entity.
3. The relevant tax regimes for each country’s tax include:

<table>
<thead>
<tr>
<th>Region</th>
<th>Country</th>
<th>Tax Regime</th>
<th>Tax</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMEA</td>
<td>United Kingdom</td>
<td>GB VAT</td>
<td>GB VAT</td>
</tr>
<tr>
<td>EMEA</td>
<td>France</td>
<td>FR VAT</td>
<td>FR VAT</td>
</tr>
<tr>
<td>EMEA</td>
<td>Germany</td>
<td>DE VAT</td>
<td>DE VAT</td>
</tr>
<tr>
<td>APAC</td>
<td>Australia</td>
<td>AU GST</td>
<td>AU GST</td>
</tr>
<tr>
<td>APAC</td>
<td>Singapore</td>
<td>SI VAT</td>
<td>SI VAT</td>
</tr>
<tr>
<td>APAC</td>
<td>New Zealand</td>
<td>NZ VAT</td>
<td>NZ VAT</td>
</tr>
<tr>
<td>NAM</td>
<td>United States</td>
<td>US SALES TAX</td>
<td>• US STATE SALES TAX</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• US COUNTY SALES TAX</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• US CITY SALES TAX</td>
</tr>
<tr>
<td>NAM</td>
<td>Canada</td>
<td>CA HST &amp; GST</td>
<td>• CA HST</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• CA GST</td>
</tr>
</tbody>
</table>

**Setting Up Tax Configuration Options**

1. On the Create Legal Entity Tax Profile page select EMEA LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>GB VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>Blank</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Create Another.

2. Select GB LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>GB VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>EMEA LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>
Click Save and Create Another.

3. Select FR LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>FR VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>EMEA LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Create Another.

4. Select DE LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>DE VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>EMEA LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Create Another.

5. Select APAC LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>AU GST</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>Blank</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Create Another.

6. Select AU LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>AU GST</td>
</tr>
</tbody>
</table>
Define Common Procurement Configuration

<table>
<thead>
<tr>
<th>Configuration for Taxes and Rules</th>
<th>Parent first party with party overrides</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>APAC LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click **Save and Create Another**.

7. Select SI LE in the **Legal Entity** field. In the **Configuration Options** tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>SI VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>APAC LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click **Save and Create Another**.

8. Select NZ LE in the **Legal Entity** field. In the **Configuration Options** tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>NZ VAT</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>APAC LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click **Save and Create Another**.

9. Select NAM LE in the **Legal Entity** field. In the **Configuration Options** tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>US SALES TAX</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Party-specific configuration</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>Blank</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click **Save and Create Another**.
10. Select US LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>US SALES TAX</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>NAM LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Create Another.

11. Select CA LE in the Legal Entity field. In the Configuration Options tab enter:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Regime Code</td>
<td>CA GST &amp; PST</td>
</tr>
<tr>
<td>Configuration for Taxes and Rules</td>
<td>Parent first party with party overrides</td>
</tr>
<tr>
<td>Configuration for Product Exceptions</td>
<td>Parent first party organization</td>
</tr>
<tr>
<td>Parent First Party Organization</td>
<td>NAM LE</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>01-Jan-01</td>
</tr>
</tbody>
</table>

Click Save and Close.

FAQs for Define Tax Regimes

**What's a service subscription?**

A service subscription is the setup for provider services. It applies to the transactions of one configuration option setup for a combination of tax regime and legal entity or business unit. Oracle Fusion Tax lets you use the tax services of external service providers for tax calculation of US Sales and Use Tax on Oracle Fusion Receivables transactions.

You can use the tax services of these external service providers:

- Taxware, LP: a First Data Company
- Vertex, Inc.

If you integrate with a tax service provider, these actions are not required for Receivables transactions:

- Entering tax classification codes on transaction lines.
- Entering transaction line attributes in the Additional Tax Determining Factors region.

Tax service provider integration returns the calculated tax lines to Oracle Fusion Tax. The tax lines for Receivables transactions returned by tax service providers are stored in Oracle Fusion Tax similar to the way tax lines calculated by the application itself are stored.
Why are controls and defaults important?

Throughout Oracle Fusion Tax care is taken to minimize your effort in creating setup. One way of doing this is the extensive use of defaulting so that you can enter your data once and use the defaults that appear on the subordinate or child records where applicable. For example, many values you enter on the tax regime appear as defaults on each tax that is associated to that tax regime. Generally, you can override the data where necessary if the defaulted value is not correct.

Also, to ensure maximum flexibility, as well as to ensure that the accuracy and integrity of the data and transactions are maintained, Oracle Fusion Use Tax makes extensive use of data-driven controls that enable and control how tax functionality works. For example, you have the requirement to set up tax recovery for value-added tax (VAT) processing. Enable the Allow tax recovery option on the tax record so you can set up tax recovery rates for this type of tax.

Manage Payment Terms

Payment Terms: Explained

Payment terms are used to automatically create installments on an invoice with up to three levels of discount. You can define payment terms to create multiple installments and multiple levels of discounts. Share payment terms across business units through set assignment.

Payment terms consist of one or more lines, each of which creates one invoice installment. Each payment term line and corresponding installment have a due date and up to three discount dates. Each payment term line and corresponding installment also have due or discount amounts. When you define payment terms, you specify either percentages or fixed amounts.

This figure shows the components of a payment term. Each payment term consists of one or more lines, and each line can have up to three discounts. Assign payment terms to one or more sets to share them across business units.
Important
If you update the payment terms on an invoice, Oracle Fusion Payables immediately recalculates the installments for the invoice. You must re-enter any manual adjustments you made to the previous installment.

Payment Terms Due Dates and Discount Dates
Payment terms due dates and discount dates are based on one of the following:

- **Fixed Date**: A specific day, month, and year when an installment is due for payment.
- **Days**: A number of days added to the invoice terms date.
- **Calendar**: A Payables calendar that is divided into periods. Assign a due date to the period that includes the invoice terms date. You can assign due dates to avoid weekends, holidays, and so on. You cannot assign calendar-based terms to an invoice if a period is not defined for the terms date.
- **Day of Month**: A type of payment term with the following attributes:
  - **Day of Month**: A specific day of the month when an installment is due for payment. For example, enter 15 to schedule payment on the fifteenth day of the month. Enter 31 to schedule payment for the last day of the month, including months with less than 31 days.
  - **Cutoff Day**: The day of the month after which the installment due dates and discount dates advance to a future month. If you do not
specify a cutoff day, the current accounting month is used to determine due dates and discount dates.

- **Months Ahead**: If you enter 0 and the invoice terms date is the same as, or later than, the cutoff day, Payables uses the day of the month in the next month to set the installment due date.
  
  For example, if the **Cutoff Day** is 11, **Day of Month** is 15, and **Months Ahead** is 0, and you enter an invoice with a terms date of January 12, the installment due date is February 15. If **Months Ahead** is 1, the installment due date is March 15. If the **Cutoff Day** is 11, **Day of Month** is 15, and **Months Ahead** is 0, and you enter an invoice with a terms date of January 10, the installment due date is January 15.

**Note**

Only due dates, not discount dates, can be based on a calendar.

**Default Payment Terms**

If you enter an **Identifying PO** on an invoice, the purchase order provides the default payment terms. If you do not enter an **Identifying PO**, the supplier site provides the default payment terms. If there are no payment terms for the supplier site, the payment terms from the Manage Invoice Options page are used. You can override the default payment terms on any invoice.

This figure shows the payment term defaulting flow during invoice entry.
Manage Units of Measure

Units of Measure, Unit of Measure Classes, and Base Units of Measure: How They Fit Together

Define units of measure, unit of measure classes, and base units of measure for tracking, moving, storing, and counting items.

The Quantity unit of measure class contains the units of measure Box of 8, Box of 4, and Each. The unit of measure Each is assigned as the base unit of measure.

Unit of Measure Classes

Unit of measure classes represent groups of units of measure with similar characteristics such as area, weight, or volume.

Units of Measure

Units of measure are used by a variety of functions and transactions to express the quantity of items. Each unit of measure you define must belong to a unit of measure class.

Base Units of Measure

Each unit of measure class has a base unit of measure. The base unit of measure is used to perform conversions between units of measure in the class. For this reason, the base unit of measure should be representative of the other units of measure in the class, and should generally be one of the smaller units. For example, you could use CU (cubic feet) as the base unit of measure for a unit of measure class called Volume.
Assigning Base Units of Measure to Unit of Measure Classes: Examples

Each unit of measure class must have a base unit of measure.

**Scenario**
This table lists examples of unit of measure classes, the units of measure in each unit of measure class, and the unit of measure assigned as the base unit of measure for each unit of measure class. Note that each base unit of measure is the smallest unit of measure in its unit of measure class.

<table>
<thead>
<tr>
<th>Unit of Measure Class</th>
<th>Units of Measure</th>
<th>Base Unit of Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>dozen</td>
<td>each</td>
</tr>
<tr>
<td></td>
<td>box</td>
<td></td>
</tr>
<tr>
<td></td>
<td>each</td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td>pound</td>
<td>gram</td>
</tr>
<tr>
<td></td>
<td>kilogram</td>
<td></td>
</tr>
<tr>
<td></td>
<td>gram</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>hour</td>
<td>second</td>
</tr>
<tr>
<td></td>
<td>minute</td>
<td></td>
</tr>
<tr>
<td></td>
<td>second</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>cubic feet</td>
<td>cubic inches</td>
</tr>
<tr>
<td></td>
<td>cubic centimeters</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cubic inches</td>
<td></td>
</tr>
</tbody>
</table>

Defining Unit of Measure Standard Conversions: Examples

A unit of measure standard conversion specifies the conversion factor by which the unit of measure is equivalent to the base unit of measure.

**Scenario**
This table lists examples of unit of measure classes, one unit of measure included in each class, the base unit of measure for the unit of measure class, and the conversion factor defined for the unit of measure.

<table>
<thead>
<tr>
<th>Unit of Measure Class</th>
<th>Unit of Measure</th>
<th>Base Unit of Measure</th>
<th>Conversion Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>dozen</td>
<td>each</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1 dozen = 12 each)</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>
</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 Corporate Procurement Cards

Creating Corporate Cards: Points to Consider

There are two ways to create corporate cards. You can choose either of the following options at different points in time or you can perform both simultaneously.

- Automatic corporate card creation
- Manual corporate card creation
Automatic Corporate Card Creation

Automatic corporate card creation applies only to travel cards. Travel cards are corporate cards that are used for travel. New employees are typically handed new corporate cards, but the information on the cards is not manually entered into the application at that time. You can create corporate cards automatically by selecting an employee matching rule for new cards on the Upload Rules tab on the Create Corporate Card Program page. Then, when the corporate card transaction file containing transactions for the new card is uploaded for the first time to the application, the corporate card transaction upload and validation process uses the matching rule to uniquely match the new corporate card to the new employee. The application automatically enters the transaction data for the new corporate card and associates it with the applicable employee using the specified rule. If the rule fails to identify a unique match, the application leaves the corporate card unassigned. If desirable, each corporate card program can have a different matching rule.

Note
To reduce or eliminate manual effort, automatic corporate card creation is recommended.

Manual Corporate Card Creation

You can manually create corporate cards for employees in the Create Corporate Card popup where you enter the following data:

• Corporate card program
• Company account name
• Corporate card number
• Employee name and number
• Expiration date
• Maximum amount per transaction (applicable only for procurement cards)
• Maximum amount per billing period (applicable only for procurement cards)

Note
Manual creation of corporate cards is the exception, rather than the rule.

Corporate cards are company account-specific. For example, if an employee transfers to another organization within your company and the organization belongs to another company account, then you must create the corporate card again with the applicable company account name.

Define Common Payables and Procurement Options

Common Options for Payables and Procurement: Critical Choices

Common Options for Payables and Procurement are setup options that are used by features throughout the procure-to-pay business flow, such as default
accounts, additional legal entity information, accounting options, and self-billed invoices.

Set options for the following:

- Default distributions
- Offset segments
- Currency conversion
- Expense accruals
- Self-billed invoices
- Legal entity information

**Default Distributions**

Default distributions are used to define the various accounts applicable for accounting for payables transactions. Invoices may get some distributions from supplier site assignments and others from the common options setup.

**Offset Segments**

If you enter invoices for expenses or asset purchases for more than one primary balancing segment value, you might want to use automatic offsets to keep your Oracle Fusion Payables transaction accounting entries balanced. If you do not use automatic offsets, Payables creates a single liability accounting entry for invoice transactions and a single cash type accounting entry for payment transactions.

**Currency Conversion**

This table lists the options you can set for currency conversion.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Require conversion rate entry</td>
<td>Require a conversion rate whenever you enter an invoice or a payment in a currency other than your ledger currency. If you maintain daily rates, Payables populates the rate automatically based on the date and the rate type you enter. If daily rates do not exist for the date and rate type, and if this option is enabled, you cannot enter or save the transaction. If the conversion rate type is <strong>User</strong>, then Payables always requires that you enter the conversion rate. You cannot create accounting entries for, or pay foreign currency invoices without conversion rates. If you do not enable this option, after you have entered invoices or created payments, you can enter conversion rates manually or by using the Apply Missing Conversion Rates program. When you create a bills payable, you are required to enter a maturity rate, rate type, and date.</td>
</tr>
</tbody>
</table>
Conversion rate type

Default conversion rate type when you enter invoices or create payments. You can change it at invoice entry or payment creation time.

Realized Gain or Loss Distributions

Default realized gain and loss accounts for payments from each of your bank accounts. If the conversion rate changes between invoice entry and payment, the application automatically calculates the realized gain or loss and records it in these accounts.

### Expense Accruals

Determine when to accrue for expense items.

### Self-Billed Invoices

This tables lists the options you can set for self-billed invoices.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gapless invoice numbering</td>
<td>Enable gapless, that is, no breaks in numbering, invoice number generation for your buying organization during pay on receipt processing. You can enable gapless numbering for the entire business unit with this setting or limit it to a supplier site.</td>
</tr>
<tr>
<td>Buying Company Identifier</td>
<td>A unique identifier for the business unit that is included in the invoice number created by the pay on receipt process and in the debit memo number resulting from return receipts.</td>
</tr>
</tbody>
</table>

### Legal Entity Information

This tables lists the options you can set for legal entity information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAT Registration Member State</td>
<td>If your company operates in a member state of the European Union, select the name of the country.</td>
</tr>
<tr>
<td>VAT Registration Number</td>
<td>If your company operates in a member state of the European Union, enter the value-added tax (VAT) registration number for your organization. Your organization is assigned a VAT Registration Number if you register to pay VAT. The first two characters of a VAT registration number are the country code for the country or state where the registered company or organization is located.</td>
</tr>
<tr>
<td>Bill-to Location</td>
<td>Enter the bill-to location to provide default values. The application uses the Bill-to Location to derive legal entity information.</td>
</tr>
</tbody>
</table>
Define Purchasing Configuration

Define Common Purchasing Configuration

Purchase Order Line Types: Examples

These examples demonstrate why the line type feature is an important part of the purchase order. It enables you to clearly differentiate orders for goods from those for services.

**Quantity-Based Purchasing**
Use quantity-based line types when you want to specify the quantity, unit of measure, and unit price for the items you are ordering. Oracle Fusion Purchasing provides Goods as an initial quantity-based line type. You can modify this line type or create new quantity-based line types to satisfy your business needs.

<table>
<thead>
<tr>
<th>Type</th>
<th>Item</th>
<th>Quantity</th>
<th>UOM</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goods</td>
<td>A554888</td>
<td>8</td>
<td>Each</td>
<td>$1,107</td>
</tr>
</tbody>
</table>

**Fixed Price Services Purchasing**
You can use fixed price-based line types when you want to order general business services by a fixed amount. Oracle Fusion Purchasing provides Fixed Price Services as an initial fixed price-based line type. You create an order for fixed price service by selecting a fixed price services line type, category, item description, and total amount of the service. You can receive and match fixed price services by amount.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Category</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed Price Services</td>
<td>Office Cleaning</td>
<td>Office Miscellaneous</td>
<td>$1,350</td>
</tr>
</tbody>
</table>

Create Document Style: Critical Choices

Purchase order document styles allow organizations to control the look and feel of the purchasing document in the application to match its business usage.
Through reusable document styles, organizations can turn on or off various procurement features, thereby simplifying the user interface. In addition, document styles provide the ability to define purchasing document names that align more closely with the naming conventions of your organization's business. When a purchasing document is created using a document style disabled features are hidden. For example, if price breaks are not allowed on the document style then agreements using this style will not display the price break region.

**Commodities**

You can create a document style for a specific commodity such as services. This document style optimizes field labels and presentation for that commodity, thereby simplifying purchase order entry.

**Blanket Purchase Agreement**

You can enable a document style for use with blanket purchase agreements. This document style could be used to limit the use of price breaks or customize the document name. For example, a construction company might name their agreement styles Equipment Agreement and Supply Agreement to easily separate the two types of agreements.

**Contract Purchase Agreement**

You can enable a document style for use with contract purchase agreements utilizing the naming convention of your organization.

**Change Order Template: Explained**

A change order template is a set of guidelines that enables an organization to specify what constitutes an internal change and what constitutes an external change to a procurement document during the course of its lifecycle. Some of the document change terminology:

1. **External Change Order**
2. **Internal Change Order**

**External Change Order**

This is a type of change order that modifies an attribute or attributes that may be relevant to the supplier as defined in the change order template. Examples include changes to price, amount, or contract terms. This is also referred to as a supplier facing change order. In commercial organizations these types of changes are referred to as an amendment and in a Federal organization they are called MODs or modifications.

**Internal Change Order**

This is a type of change order that modifies an attribute or attributes that may not be relevant to the supplier as defined in the change order template. Examples
includes changes to a descriptive flexfield or a certain category of attachments. These are also referred to as administrative changes.

Typically this will be a buyer or requester requested change order. That is a change order requested by a user whose role is either buyer or requester.

FAQs for Define Common Purchasing Configuration

**What’s a document style?**

Purchase order document styles allow organizations to control the look and feel of the application to match the usage of the purchasing document. Through reusable document styles, organizations can turn on or off various document attributes, thereby simplifying the user interface to fit their business needs.

Define Procurement Configuration Options

**Price Break Type: Critical Choices**

You can select the price break type that defaults on blanket purchase agreements from the Configure Procurement Business Function page.

**Cumulative Pricing**

Select Cumulative Pricing if you want to choose the price break by adding the current release shipment quantity to the total quantity already released against the purchase agreement line.

---

**Note**

Cumulative Pricing can not be used with global agreements.

---

**Non-Cumulative Pricing**

Select Non-Cumulative Pricing if you want to choose the price break by using the individual release shipment quantity of the agreement line.

Choosing an Inventory Organization: Explained

Purchasing documents can be created to replenish goods stocked in an organization’s inventory. You can associate each of your procurement business units with one inventory item master organization from the Configure Procurement Business Function page.
Choosing an Inventory Organization

When you associate your procurement business unit (BU) with an inventory organization, items you define in this BU become available throughout procurement. Do not change the Inventory Organization after you have already assigned one to a procurement BU.

Allow Retroactive Pricing: Critical Choices

Retroactive price updates automatically update existing purchase orders retroactively with price break quantities from the parent blanket agreement.

Open Orders Only

Choose this option to allow retroactive price updates to open orders with no receiving or invoicing activity.

All Orders

Choose this option to allow retroactive price updates to all orders irrespective of whether they have been received or invoiced.

Receipt Close Point: Explained

Used to select the close point, which is when the shipment is closed for receiving:
You can override this option for specific items and orders.
Selecting the close point gives you more precise control over when the shipment is accounted for.

Note
The receipt close tolerance percentage must be set in combination with this setting.

Accepted

Ordered goods have passed inspection and are ready for use.

Delivered

Ordered goods have been delivered and are ready for use.

Received

Ordered goods have been received and are ready for use.
**Match Approval Level: Critical Choices**

A transaction status match combination after which the document may be considered approved for payment.

---

**Note**

The invoice match option in the purchase order schedule and the match approval level described here are independent options. The invoice match option determines whether Payables performs invoice matching to the purchase order or the receipt.

---

**Two-Way**

Purchase order and invoice quantities must match within tolerance before the corresponding invoice can be paid.

**Three-Way**

Purchase order, receipt, and invoice quantities must match within tolerance before the corresponding invoice can be paid.

**Four-Way**

Purchase order, receipt, accepted quantities from inspection, and invoice quantities must match within tolerance before the corresponding invoice can be paid.

---

**Group Requisitions: Critical Choices**

During automatic creation of purchase orders from requisitions use the following options to control how requisition lines are combined on order lines in the new document.

**Group Requisitions**

By default the requisitions are converted into purchase order lines individually. This option allows consolidation of requisition lines from across multiple requisitions into a single purchase order.

Select this checkbox to group requisitions into a single purchase order. The application will try to group all the requisitions which share the same:

- Requisitioning BU and Sold-to BU
- Document style
• Supplier and site
• Currency and conversion rates
• Buyer

**Group Requisition Lines**

By default the requisition lines are combined into individual lines based on document type. This option determines whether each requisition line being fulfilled in the order will have its own order line or can be combined with other requisition lines. Select this checkbox to group requisition lines into the same purchase order line. The application will try to group requisition lines which share the same:

• Line type
• Item, item revision, item description, supplier item number, supplier config ID
• Category
• UOM
• Source agreement and source agreement line
• Need-by date (used optionally if the Use need-by date checkbox is selected)
• Ship-to organization and location (used optionally if the Use ship-to organization and location is selected)

For all requisition lines being grouped into the same purchase order line, the application will further group these lines into schedules if they share the same:

• Need-by date, ship-to location and ship-to organization
• Destination type

**Standard Purchasing Terms and Conditions: Explained**

Standard terms and conditions outline any legal or functional constraints under which the purchase order or purchase agreement will be conducted. These terms and conditions apply to any orders executed by the procurement business unit and can be made available in all installed languages. Use the Define Purchasing Terms and Conditions window to enter standard terms and conditions for purchasing documents.

Standard terms and conditions are included with each purchasing document sent to the supplier.

**Terms**

Purchasing terms commonly indicate the buying organization’s rules and expectations as related to pricing and payment.
Conditions

Purchasing conditions typically describe the buying organization’s rules related to but not limited to delivery, acceptance of delivery, cancellations, additions to the approved order, and general behavior of the supplier during the course of the transaction.

Languages

Indicate that this set of terms or conditions are not to be made available in all installed languages by selecting “Disable terms and conditions for all languages.”

FAQs for Configure Procurement Business Function

What's a receipt close tolerance percent?

This setting is used in combination with the closing point to close a schedule for receiving. It is the allowable difference (expressed as a percentage) between the quantity stated on the order schedule and the actual quantity received. Quantities can vary up to this difference without preventing an automatic closed for receipt status. You can override this option for specific items and orders.

Note

You must also set the receiving close point.

For example, with a receipt close tolerance percent of 99 with a quantity ordered of 100 and a closing point of receipt the schedule would automatically be closed for receiving when 1 of the 100 are received.

What's a contract terms layout?

For purchasing documents select a contract terms layout for the document type. These templates determine what information is displayed along with providing the headers, footers, text style, and pagination of the printed document. You can create your own custom layout in RTF format and upload it to Oracle BI Publisher or select one already provided.

Configure Requisitioning Business Function

Configure Requisition Business Function: Explained

The Procurement Application Administrator has access to the Configure Requisition Business Function page for setting up a business unit that has a
requisitioning business function associated with it. The attributes specified here are used to default values and behavior of the application when users are creating requisitions and purchase orders for the requisitioning BU.

**Requisitioning Section**

**Next Requisition Number**

The Next Requisition Number is used to specify the next number to be used when generating a requisition. When a requisition is created online, the Next Requisition Number is assigned to the requisition; the number specified cannot be in use by an existing requisition. Note that when a requisition is created through the requisition import process, a numeric or alphanumeric requisition number can specified on the requisition record; it will be accepted if there is not in use by an existing requisition number.

**Default Deliver-To Organization**

The default organization is used as the deliver-to organization for a requisition line if it is a global location. This organization is used to derive the list of item master items that are accessible to the user when creating a purchase order for the requisitioning BU.

**Line Type**

The Line Type is the value specified to be defaulted on requisition lines created for the requisitioning BU. Line Type can be modified.

**One-Time Location**

The One-Time Location is the location code to be defaulted as the deliver-to location for the requisition line when the requester specifies a one-time delivery address on a requisition. The location specified must be a global location that is enabled for the requisitioning BU.

**Reapproval required for changes made during an active approval process**

Reapproval required for changes made during an active approval process is applicable when allowing approvers to modify a requisition when it is routed for approval. It controls whether the requisition must be sent back for reapproval when the approver submits the modified requisition.

**Group Requisition Import**

The Import Requisition process can be used to import requisitions from other Oracle or non-Oracle applications. On import, requisition lines are grouped first by requisition header number, then by the provided Group Code, then by the value set in the Group-by input parameter (None, Buyer, Category, Item, Location, or Supplier). The specified attribute is used as the default value for Group-by. All remaining requisition lines that have not yet been assigned a requisition number will be grouped together under the same requisition.

**Create Orders Immediately for Requisition Import**

Create orders immediately after requisition import controls whether the Generate Orders program will run immediately after the requisition import process is complete.
Purchasing News

The contents specified in Purchasing News is displayed in the Purchasing News section on the Shop Home page. If the URL and URL display name are specified, they are displayed on the Shop Home page for the requesters to drill down and view more information.

Context Values for Requisition Descriptive Flexfields

You can extend the attributes of a requisition at the header, line, and distribution level using Descriptive Flexfields. Specifying the context value pulls in the associated descriptive flexfields when the user enters the requisition.

Purchasing Section

Default Procurement BU

A requisitioning BU can be served by multiple procurement business units. If a procurement BU cannot be determined based on information on the requisition line, the Default Procurement BU is used to process all requisition lines.

Price Change Tolerance

The Price Change Tolerance is applicable when there is a price change on the purchase order line associated with a requisition line. If the value is null, no checks will be performed. If the value is a valid numeric value, then any changes made to the price on the purchase order line must be within the tolerance percentage value, or the purchase order cannot be submitted. The tolerance can be specified using the tolerance percentage or tolerance amount. The more restricting of the two tolerances will take precedence if both are specified.

Ship-to Location

When the purchase order cannot derive a ship-to location, the specified Ship-To on the Requisitioning BU is defaulted.

Cancel Backing Requisitions

Cancel Backing Requisitions controls whether a backing requisition should be canceled when there is purchase order cancellation.

Options are:

- Always: When canceling the purchase order, Oracle Fusion Purchasing also cancels the requisition.
- Never: When canceling the purchase order, Oracle Fusion Purchasing does not cancel the requisition, therefore it is available for inclusion on another purchase order.
- Optional: When canceling the purchase order, the buyer is given the option to cancel the requisition.

Allow Requester-To-Agreement UOM Conversion

If a requisition does not have an agreement specified, Allow requester-to-agreement UOM conversion is used to specify whether Requisition UOMs can be converted to Agreement UOMs during agreement sourcing. Checking this box
indicates that agreements that meet the sourcing criteria, but have Agreement Line UOMs different from Requisition Line UOMs, can be considered during agreement sourcing. If the box is left unchecked, such agreements will not be considered.

**Define Procurement Agents**

**Agent Security: Explained**

Use the Create Procurement Agent page to control a procurement agent's access to procurement activities for a business unit. You can implement document security for individual document types. You can also control a procurement agent’s access to activities such as suppliers, approved supplier list entries, and business intelligence spend data through the settings on this page.

**Implementing Document Security**

The key elements for document security are the procurement business unit, enabling agent access to document types, and the access levels to other agents’ documents.

**Create Procurement Agent: Critical Choices**

The Create Procurement Agent page is used to create or edit a procurement agent and define that agent’s access to procurement functionality within a procurement business unit.

**Note**

The following Fusion predefined roles are controlled by procurement agent access configuration: Buyer, Category Manager, Procurement Manager, Procurement Contracts Administrator, Supplier Administrator, and Catalog Administrator.

**Procurement BU**

Assign the agent to one or more procurement business units.

**Action**

Enable the agent to access one or more procurement action for each procurement business unit.

- Manage Requisitions
• Manage Purchase Orders

• Manage Purchase Agreements: Enable access to blanket purchase agreements and contract agreements.

• Manage Negotiations: Enable access to Sourcing negotiations, if implemented by your organization.

• Manage Catalog Content: Enable access to catalog content. This action allows an agent to add or update blanket purchase agreement line information as part of the collaborative catalog authoring.

• Manage Suppliers: Enable access to create and update supplier information.

• Manage Approved Supplier List Entries: Enable access to create and update approved supplier lists.

Access to Other Agents’ Documents

Assign an access level to documents owned by other procurement agents for each procurement business unit. Note that an agent can perform all actions on their own documents as long as they have procurement BU access.

• None: The agent cannot access documents owned by other agents.

• View: Permits the agent to search and view other agents’ documents.

• Modify: Permits the agent view, modify, delete, and withdraw other agents’ documents.

• Full: Permits the agent full control of other agents’ documents which include view, modify, delete, withdraw, as well as perform document actions including freeze, hold, close, cancel, and finally close.
Define Self Service Procurement Configuration

Manage Information Template Descriptive Flexfields

Information Template: Explained

An information template is used to gather additional information from a preparer. It can be assigned to an item, a category, or a smart form. Information templates are used in the creation of a Smart Form to provide the flexibility to add additional attributes in a smart form in order to gather required information from a preparer. Information templates are also applicable to item master items and purchasing categories.

The data entered for an information template, which is associated with a smart form, item or category, is available as attachments in downstream products (such as Purchasing) after the requisition is approved. When creating an information template, the catalog administrator selects the attachment category that determines if the attachment will be available to the supplier or buyer.

Using Information Templates

Information Templates are created in a Procurement Business Unit and are available to Requisitioning Business Units serviced by that Procurement BU. In the event where a Requisitioning BU is serviced by multiple Procurement BUs, and more than one service provider had assigned an information template to an item or category, applicable information templates from all service provider Procurement BUs will be returned.

Information templates are available to the preparer if the items or smart forms that the information templates are associated with are available to the preparer.

Procurement catalog administrators can define a unique information template name so they are easily identifiable in a smart form. Information template header information provides users the ability to specify a non-unique Display Name, while creating information templates with unique information template names. For example, more than one procurement BU can maintain information templates to collect business card information. The same Display Name, Business card information, can be used on these information templates to indicate the purpose of these templates when displayed in Oracle Fusion Self
Oracle Procurement Cloud Implementing Procurement

Service Procurement. Procurement Catalog Administrators can also define an information template section description or instruction text providing preparers with specific instructions on how to fill out the form.

Information templates can only be deleted if they are not referenced. An information template is considered referenced if it is applied on any requisition lines, whether in completed or incomplete state. This is to prevent deletion of an information template that is currently in use.

Once an information template is deleted, it is no longer returned on the Manage Information Templates page.

Adding Attributes

Information template attributes are maintained as Descriptive Flex Fields (DFFs). Attributes first need to be setup in the Descriptive Flexfields application, and the catalog administrator specifies the DFF context on the Create and Edit Information Template page to apply the list of attributes.

For example, the catalog administrator set up a context Business Cards Marketing, with the following context sensitive fields:

- Job Title
- Organization
- Office Location

When creating an information template, the catalog administrator can then specify in the Attribute List field the context Business Cards Marketing, which will associate the attributes to the information template.

Note

The maximum number of attributes that can be created for an information template is fifty.

Existing information attributes are maintained as attachments downstream, such as in Purchasing.

Supported Attributes

The following attribute types are supported by DFFs:

- Text: Text attributes can be setup using DFF to be added to an information template. For example, the procurement catalog administrator can create an information template called Business Card Information USA to be used for collecting related information when ordering business cards. Examples of text typed fields are Name, Title, Address and so on.

- Number: Procurement catalog administrators can create number typed attributes using DFFs, to be used in an information template. For example, Zip Code, Telephone, and Area Code.

- Standard Date, Standard DateTime and Time: Procurement catalog administrators are able to define date format attributes in DFFs, to be used in an information template. This allows for automatic date formatting according to globalization requirements, since 09/01/2007 may mean September 1, 2007 in the US, but January 9, 2007 in others.

- List of Values: Value sets can be added to Information Templates through DFFs as List of Values. Implementing attributes as List of Values allows enforcement of values that can be populated in these fields. For example, as part of an address, the Country field can be implemented as list of values (LOV) containing only countries that are applicable.
• Choicelists: Choicelists make use of value sets as well, similar to List of Values.

End Dates

Procurement Catalog Administrators can specify an End Date on an information template. An information template is inactive if the system date is more than or equal to the End Date.

When an information template is inactive, it will no longer be applied when items (to which this information template is assigned) are added to the requisition. Requisitions created with lines that are associated to this information template will continue to display the information template information.

For incomplete requisitions, the inactive information templates are no longer available at the time the requisition is retrieved.

For copied and withdrawn requisitions, information templates are also no longer available if the information template is inactive at the time the requisition is copied or resubmitted.

Manage Catalog Category Hierarchy

Catalog Category Hierarchy: Overview

Category hierarchy presents a hierarchical view of the catalog to users. Category hierarchies allow administrators to create a parent category that includes other categories, which are known as child categories. When users navigate through the parent category, the child categories appear, helping users to navigate quickly to the category which contains the products they need.

Category Hierarchy: How Browsing Categories and Item Categories Fit Together

If you manage a large number of products and services you may need a mechanism to organize the products in the catalog to make it easier for users to navigate to the products they want to buy. The category hierarchy presents a hierarchical view of the catalog to users.

Category hierarchies allow you to create a parent category that includes other categories, which are known as child categories. When users navigate through the parent category, the child categories appear, helping users to navigate quickly to the category which contains the products they need. Categories are used to classify items.

You can develop your own method of categorizing products or you can use standard coding systems such as UNSPSC. Some of the benefits of adopting a standard coding system are visibility of spend analysis throughout the corporation, cost control optimization, and ability to explore all the ecommerce capabilities.

The figure below shows the category hierarchy for a catalog. There are two types of categories in the catalog that define a catalog hierarchy: Browsing categories
and item categories. It is not required to have the same number of levels in all branches of the hierarchy.

### Browsing Categories

Browsing categories are also known as navigation categories. They define the category hierarchy for category browsing. The category hierarchy helps users browse for catalog items. Browsing categories can be either a parent or child to another category, but cannot contain any items. Browsing categories are optional and companies can decide what categories should be enabled for browsing.

You can associate catalogs (local, punchout, informational) and smart forms to the browsing categories. When user navigates to the category, the associated content type will be displayed. An alternative to setting up browsing categories is to tag punchout, informational, and smart forms with keywords, so that users can find them when performing basic search.

### Item Categories

Item categories are used to group items for various reports and programs. A category is a logical classification of items that have similar characteristics. For Procurement, every item must belong to an item category. Item categories allow you to:

- Place item categories under browsing categories.
- Search the catalog and sort by item category name. The item category name is displayed in the search pages.
- Bulkload by item category code.

### Category Hierarchy with Catalog Association: Explained

Users can search for all content (local content, punchout, smart forms, informational content) regardless of how the content is grouped. Administrators can group punchout, informational catalogs, and smart forms by category and the browsing feature will also retrieve punchout, informational catalogs, and smart forms together with local content.
Local content (item master items and agreement lines) is associated with purchasing categories. Smart forms, punchout, and informational catalogs can optionally be associated with any level of the category hierarchy (browsing or purchasing category).

**Hierarchy With Associated Catalog Content**

When the user associates the punchout, informational, local, and smart form to a category, the system travels up and down the tree to associate the punchout, informational, local and smart form with all the browsing and purchasing categories of the same branch. Item master items, and agreement items are indexed with their corresponding purchasing categories. For example, in the illustration below, when the user navigates down the branch from Information Technology browsing category to the Computer Servers purchasing category, the search results will always include the Dell USA punchout which is associated with Computers. The system associates the punchout catalog Dell USA with the categories of the same branch as Computers which are Information Technology, Components for Information Technology, Computers, and Computer Servers.

The informational catalog How to Request Computer Services is associated with the browsing category Information Technology. As the user navigates the branch of Information Technology, the Informational Catalog is seen at the level of Information Technology, Components for Information Technology, System Cards, Computers, Memory Module Cards, and Computer Servers.

Local catalog items also show up during browsing. Using the example in the figure below, items in BPAs with suppliers Techworks or Zones Corporate that are tied to the purchasing categories Memory Module Cards or Computer Servers will show up as the user navigates down the Information Technology branch, based on the content available to the user via content zone.

The procurement catalog index is automatically updated after any changes to the hierarchy are saved.

The figure below shows catalog category hierarchy structure.
Define Supplier Portal Configuration

Manage Supplier User Roles

Supplier User Provisioning : How It Works

Supplier User Provisioning refers to the process of establishing suppliers with access to Oracle Fusion Supplier Portal. It enables the buying organization to create and maintain user accounts, job roles, and data access controls for supplier contacts. An important part of supplier user provisioning is to provision job roles, which give users the ability to perform on-line business tasks and functions with the buying organization which are associated with their job.

The content Supplier Users can access and tasks they can perform are tightly controlled by the buying organization. However, a key feature of Oracle Fusion Supplier Portal allows supplier users to assume the responsibility for user account management on behalf of the buying organization by creating and maintaining user accounts for their fellow employees that require access to the Supplier Portal application. The buying organization maintains control by granting provisioning access to their trusted suppliers, significantly reducing their administrative burden.

User Provisioning Job Roles

The seeded job roles that can perform supplier user provisioning are:

- **Supplier Administrator:** This is an internal job role to the buying organization. Users with this role are responsible for maintaining supplier profile information as well as administering user accounts for supplier contacts.

- **Supplier Self Service Clerk (SSC):** This is a supplier job role. Supplier users with this role can maintain contact profiles and request user accounts for their fellow employees. All contact profile updates and user account requests made by the SSC require approval by the buying organization.

- **Supplier Self Service Administrator (SSA):** This is a supplier job role. Supplier users with this role can maintain contact profiles and provision user accounts to their fellow employees, without requiring buying organization approval.
There are several flows in which Supplier Administrators perform user provisioning:

- Supplier registration review and approval.
- Supplier contact change request review and approval.
- Suppliers work area, Edit Supplier flow where supplier contacts are maintained.

In each of these flows the Supplier Administrator is able to create a user account, assign job roles and set data security access for a supplier contact.

**Manage Supplier User Roles Setup Page**

The Manage Supplier User Roles setup page is used by the buying organization to define the job roles that can be provisioned to supplier users for accessing Oracle Fusion Supplier Portal. This page also controls options for how the supplier job roles are used in the various provisioning flows. These two distinct setup tasks are intended to be performed by two different internal job roles.

The Manage Supplier User Roles page serves two important setup tasks:

1. The core task is to define the list of roles that can be provisioned to supplier users in Oracle Fusion Supplier Portal provisioning flows. The supplier roles are added from the central Oracle LDAP roles repository which stores all Oracle Fusion application job roles. Once the role is added to the table, it is immediately available for provisioning to supplier contacts by the Supplier Administrator. Only the IT Security Manager job role can add and remove roles to avoid the risk of adding an internal application job role inadvertently which could result in suppliers gaining unauthorized access to internal data. This security risk is the reason only the IT Security Manager has the privilege to manage the list of supplier job roles that can be provisioned.

2. Define the supplier role usages. The Procurement Application Administrator is responsible for this setup task, which manages settings for how the supplier job roles are exposed in provisioning flows. The first column controls whether the supplier job role can be provisioned by suppliers in Oracle Fusion Supplier Portal, specifically supplier users with the SSA role.

   Additionally, default roles can be established which expedite supplier user account requests by allowing the buying organization to identify the minimum set of job roles that a supplier contact can be granted. This prevents approvers from having to explicitly review and assign job roles for each user account request.

   The IT Security Manager can also set supplier role usages as they can access all functions on the setup page, however this is typically performed by the Procurement Application Administrator. The Procurement Application Administrator cannot add or remove roles from the table.

   When the role default setup is done correctly, the Supplier Administrator (or approver) can review supplier contact user account requests with job roles selected based on the source of the request, and proceed to approve user account requests with appropriate role assignments.

   The three role usages relevant to supplier user provisioning include:
- Allow Supplier to Provision: If selected, the role can be provisioned by the SSA, assuming the role is also assigned to the SSA user.

- Default for Oracle Fusion Supplier Portal: If selected, the role is automatically added to supplier user requests in the provisioning flows enabled by Oracle Fusion Supplier Portal such as supplier registration and supplier profile maintenance.

- Default for Oracle Fusion Sourcing: If selected, the role is automatically added to supplier user requests generated in sourcing flows such as Create Negotiation.

A role in the table can be marked for one or more of the three usages.

The figure below shows the flow for managing supplier user roles.

The IT Security Manager and the Procurement Application Administrator access the Manage Supplier User Roles page through the following respective setup tasks in the Oracle Fusion Setup Manager, under Define Supplier Portal Configuration:

- Manage Supplier User Roles
- Manage Supplier User Roles Usages

Note

SSA users should be careful when removing roles from their account because they are not able to add additional roles to their own user account.
To ensure the SSA provisions proper roles to the supplier users in their company, users with the SSA job role are able to provision roles based on those roles checked in the Allow Supplier to Provision column and the set of roles they have already been assigned. This intersection, as depicted in the figure below, determines what roles they can grant to their fellow employees.

Set Up Supplier Roles: Examples

The following simple examples illustrate selecting and managing roles for supplier user provisioning.

Selecting Roles for Supplier User Provisioning:
Company ABC decides to expand supplier portal deployment and allow suppliers to access orders and agreements. The IT security manager navigates to the Manage Supplier User Roles page, searches for the supplier job role, supplier customer service representative. The IT security manager adds supplier customer service representative to the table. The procurement application administrator then navigates to the Manage Supplier User Roles page and sets supplier customer service representative as Default for Supplier Portal, and Allow Supplier to Provision.

Managing Default Roles and Defining Roles that the Self Service Administrator (SSA) can Provision:
Company ABC determines that all supplier users can be granted access to orders, shipments, receipts, invoices and payments information by default, but access to agreements will only be granted to select supplier users. The sales representative role will not be marked as a default role.

Company ABC recently implemented Oracle Fusion Sourcing and needs to provision the supplier bidder role to specific suppliers invited to sourcing events. The SSA should not be allowed to provision this role as it needs to be controlled by Company ABC. When supplier bidder is added to the table, Allow Supplier to Provision should not be checked, but Default Roles for Sourcing is checked.
Supplier User Account Administration: Explained

User accounts need to be provisioned to allow supplier contacts to access the Oracle Fusion Supplier Portal application. User account maintenance is performed for a specific supplier contact under the Contacts tab. A user account is assigned roles that determine what functions a supplier contact can perform when logging into the application.

Below are Fusion flows where a user account can be requested and managed as part of a supplier contact:

- **Create Supplier Contact:** When creating a supplier contact, the administrator can also request to create a user account for the contact, request roles and grant data access.

  **Note**

  Creating a user account for a supplier contact cannot be reversed. Once a user account is created, it cannot be deleted from the system, but it can be inactivated.

  - Edit Supplier Contact: The supplier administrator can make changes to supplier contact information as well as create or maintain the user account for the contact.
  
  - Approve supplier registration request: When an approver is approving a supplier registration, the approver can create and edit supplier contacts. Since a user account is part of a supplier contact, the approver has the ability to create a user account and assign roles within this flow.

The Supplier Administrator is responsible for:

- **Creating and inactivating supplier user accounts**

  When Create User Account is selected for a contact, a request is initiated to Oracle Fusion Identity Management (OIM) to provision the account. Status is displayed to the user to communicate provisioning status during this process.

  When the process is complete, OIM sends notification to the supplier contact with the username and temporary password to access Oracle Fusion Supplier Portal. If the process fails, a notification is sent to the Supplier Administrator alerting them that a user account was not successfully provisioned for the supplier contact.

  - Assigning supplier job roles.

  The Roles subtab controls function security which determines the business objects and task flows the supplier user can access. Supplier job roles should be assigned based on the job that the contact performs within the supplier organization, such as Customer Service Representative or Accounts Receivable Specialist.

  - Assigning data access

  The Data Access tab controls data security, or which transactions the user can access for the specific business objects their job role is associated with. There are two levels of data security; Supplier and Supplier Site. By default all supplier user accounts start with supplier level, meaning they can access all transactions.
belonging to their supplier company only. For more restrictive access, Supplier Site level, limits user access to transactions for specific Supplier Sites only.

Configure Supplier Registration

Configuring Supplier Registration: Points to Consider

The Configure Supplier Registration page enables a procurement business unit (BU) to select which optional profile components to capture on registrations, which can include; address information, business classifications, product category offerings, and bank account details. The components can be enabled for internal users and supplier users. Internal users are authenticated users in the buying organization. Supplier users are prospective suppliers accessing the Supplier Registration flow. Only components selected for internal users can then be selected for supplier users.

Selecting a component for internal users adds it to the Supplier Registration Approval flow and Internal Supplier Registration which is exposed in the Sourcing Negotiation flow. Selecting a component for supplier users adds it to the external Supplier Registration flow used by prospective suppliers.

Scope

Supplier registration is deployed at a procurement BU level. The global row allows for all procurement BUs deploying supplier registration to share a common configuration. If a procurement BU requires different components in its registration flow from the global row definition, then a new configuration can be defined specific to the procurement BU by adding a new row.

Seeded Values

The Configure Supplier Registration page is seeded with one row. For this seeded row, the scope is set to Global and none of the registration components are selected for Internal Users or Supplier Users. The default value of the Review Type field is set to Registration. This seeded record can be edited, but cannot be deleted.

Review Type

If a buying organization wants to mark suppliers after they are autocreated following registration approval the Review Type field can be leveraged. This is useful to perform any additional setup steps or offline qualification checks before conducting business with the supplier. For example, setup tax profiles to support accruals processing and review the site parameters to ensure the site is ready to transact. In order to use this field, the default value of the Review Type field should be set to Registration in the Configure Supplier Registration page. This ensures all newly created suppliers from approved registrations will have the Review Type field value set to Registration. After completing the necessary setup steps, the supplier administrator can change the Review Type to None on the supplier record to indicate that any remaining supplier setup was completed.
Registration URL Encryption

When a prospective supplier saves the registration with the intent of completing it later, the application sends an e-mail to the prospective supplier containing the URL to be used to return to the registration. The URL contains an identifier which is encrypted using an encryption key. This is done to prevent someone from altering the URL in order to gain access to registrations submitted by other companies.

If it is suspected that registrations have been tampered with, the Procurement Application Administrator can regenerate the encryption key. Once the registration key is regenerated, the registrations which were saved for later will no longer be accessible to the prospective suppliers.

Accessing Supplier Registration

A supplier registration URL for each procurement BU deploying a registration flow needs to be published, for example on a corporate web site page focused on supplier information. The URL contains a parameter for the procurement BU which navigates the user to the registration flow for that particular BU.

The registration URL for each procurement BU can be found on the Procurement Options page in the Supplier Registration URL field.

In the registration flow, the procurement BU name is displayed in the sub-branding region. To turn off displaying the procurement BU name in the sub-branding region, append the parameter showBU=N to the registration URL. Here is an example of the registration URL with showBU parameter turned off:

http://<server>:<port>/supplierPortal/faces/PrcPosRegisterSupplier?
prcBuld=<Buld>&showBU=N

Supplier Products and Services Categories : Explained

Products and services categories can be captured as part of supplier profile which can be used to identify suppliers to invite to sourcing negotiations. The categories are presented in a tree based hierarchy for easy selection, which is available in all flows where supplier profile information is captured, including: supplier registration, internal supplier registration, supplier registration approval, and the supplier master where profiles are maintained by internal supplier administrators.

The hierarchy is defined and maintained by accessing the Manage Supplier Products and Services Category Hierarchy setup task found under the Define Supplier Configuration setup activity in the Fusion Setup Manager. The buying organization may, at times, decide to reorganize categories exposed to suppliers for selection due to internal reorganization or changes in the business.

Products and Services Categories

The buying organization can collect information about the products and services categories supplied by the company on the registration flow by enabling Products and Services using the Configure Supplier Registration flow. Prospective suppliers can select the categories they are capable of supplying.
during the registration flow. If the category manager is registering a company while creating or editing a negotiation, they can also select the applicable categories on the registration.

Once the registration is submitted, it goes through an approval process in the buying organization. If the registration is approved, the Products and Services Categories is maintained by the Supplier Administrators as a part of their supplier profile maintenance task. Supplier users are able to review their Products and Services Categories in Oracle Fusion Supplier Portal as part of their supplier profile information.

The figure below illustrates the tasks which use the Supplier Products and Services Categories.

---

**Specify Supplier News Content**

**Supplier Portal Overview: Explained**

The Supplier Portal Overview provides suppliers a quick glance across transaction flows and highlights urgent tasks which are relevant to a user's job role.

The transactional tabs offer consolidated reporting views across different business objects which provide quick visibility to recent business activity. The Watchlist on the summary tab provides users with a one-stop shop for all the key tasks that need to be performed and important inquiries that need monitoring.
**Summary Tab**

The overview page serves as the home page in Oracle Fusion Supplier Portal. The page is in a tabbed structure and by default the Summary tab is selected.

The Summary Tab contains the following regions designed to alert suppliers on the latest transaction information as well as communicate general news and provide access to supplier performance reports:

- **Supplier News**: A place where relevant news is posted for supplier users. For example, server down time, upgrade notices, and so on.

- **Worklist**: A list of all notifications sent to the current supplier user. Critical transaction events generate notifications, such as communication of a new purchase order issued by the buying organization. The worklist serves as a convenient place where supplier users can view these notifications.

**Note**

An e-mail version of most notifications are also sent to the supplier.

- **Watchlist**: Contains a set of saved searches which display counts of urgent or recent transactions, possibly requiring action, such as Orders Pending Acknowledgment. Clicking a watchlist entry navigates the user to begin working on the transactions immediately.

- **Supplier Performance Reports**: Contains a list of transaction reports relevant to the user role. Report criteria can be specified for example Supplier Site, Category, or Item. Reports include:

<table>
<thead>
<tr>
<th>Report Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO Purchases Amount</td>
<td>Provides summary of PO purchase amount for the given set of parameters, which also includes supplemental information such as growth rate and percent of total.</td>
</tr>
<tr>
<td>Receipt Date Exceptions</td>
<td>Set of reports providing detailed information on Receipt Date Exceptions (comparing receipt date against the PO schedule for each receipt line to determine being early or late) for the given set of parameters. Early is defined as receiving before the promise date or need-by date if no promise date is available. Late is defined as receiving after the promise date or need-by date if no promise date is available. The Receipt Date Exception Rate provides the exceptions rate for a given time period. The lower the exception rate, the better the delivery performance. This is calculated as: Exception Amount divided by Receipt Amount, multiplied by 100.</td>
</tr>
<tr>
<td>Returns</td>
<td>Set of reports providing detailed information on returns for the given set of parameters. The report also includes growth rate, percent of total and its change for all the values.</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Invoice Amount</td>
<td>Provides a summary of invoice amount for invoices with or without a matching PO for the given set of parameters. The report also includes growth rate, percent of total and its change for all the values.</td>
</tr>
<tr>
<td>Invoice Price Variance Amount</td>
<td>Shows the variance between the invoice price and the purchase price. Payables records invoice price variances when the invoices are matched, approved, and posted. The invoice price variance is the quantity invoiced multiplied by the difference between the invoice unit price and the purchase order unit price. The report also includes growth rate, percent of total and its change for all the values.</td>
</tr>
</tbody>
</table>

**Orders Tab**

Content within this tab is primarily tailored to the role of Supplier Customer Service Representative. The tab contains the following regions:

- Orders with Recent Activity: A list of orders that are Opened, Changed, or Canceled since the specified date.

- Pending Change Orders: A list of change orders initiated by the supplier company that are either requiring supplier action or are pending review by the buying organization.

**Agreements Tab**

Content within this tab is primarily tailored to the role of Supplier Sales Representative whose main function is to manage agreements. The tab contains the following regions:

- Agreements with Recent Activity: A list of agreements that are Opened, Changed, or Canceled since the specified date.

- Pending Change Orders: A list of agreements initiated by the supplier company that are either requiring supplier action or are pending review by the buying organization.

- Pending Authoring: A list of draft agreements that are transferred to the supplier for catalog authoring. Catalog authoring is the process by which
suppliers enter or upload their catalog items to the agreement for review and approval by the buying organization.

**Schedules Tab**

Content within this tab is primarily tailored to the role of Supplier Customer Service Representative. The tab contains the following regions:

- **Open Schedules**: A list of open purchase order schedules that are not received by the buying organization. Any overdue schedules are highlighted with an alert. Open schedules can fall into three categories:
  - Order schedules not shipped.
  - Orders schedules shipped and in transit, but no shipment notice was communicated.
  - Orders schedules shipped and in transit, which have an associated shipment notice.

- **Recent Receipts**: A list of purchase order schedules recently received.

**Negotiations Tab**

This transaction tab is tailored to the role of Supplier Bidder. Users are presented with negotiation transactions that the supplier is involved in or is invited to by the buying organization. It provides a quick summary view to easily monitor the status of supplier negotiation responses.

---

**Note**

This tab is only visible when Oracle Fusion Sourcing is implemented.
Define Sourcing Configuration

Manage Negotiation Styles

Creating a Negotiation Style: Worked Example

Negotiation styles control the definition of your negotiation documents. Negotiation styles can specify the terminology used within the document and control which processing capabilities can be performed using the style. For example, you can define a simple negotiation style and then use it to create very straightforward, streamlined negotiations. Alternately, you can create a negotiation style that takes advantage of many processing features. You can then use this style to create a complex negotiation. Using negotiation styles, you can also define default textual content for certain sections of a negotiation document.

To define a negotiation style, you select which negotiation types the style can be used with, which features are available using the style, and any default text you want to appear in the document.

Creating a Negotiation Style

1. From the Manage Negotiation Styles page, click the icon to create a new style.

2. On the Create Negotiation Style page, you define how your style can be used and which negotiation features are available using it.

3. In the Document Types region, you can specify the types of negotiations for which this negotiation style is appropriate. You can also change the terminology used for the negotiation type and the response type.

4. If you wish to add text instructions to the negotiation, click the edit icon. The instructions are visible to your suppliers and can include any directions on how to create the response.

5. In the Controls region, select which negotiation features you wish to make available using this style.

6. When finished, save your new negotiation style.
Negotiation Styles: Examples

You can create multiple negotiation styles that control the creation of your negotiation documents. Negotiation styles can specify the terminology used within the document and control which processing capabilities can be performed using the style. You can also create default content for certain sections of a negotiation document.

Specifying Document Terminology

For each negotiation there is a pair of documents: negotiation document created by the category manager and a response document created by the supplier contact. Each of these documents has a label. Within the application, there are three types of negotiations: auction, RFI, or RFQ. Each type has its own default document labels; however, you can change the labels used in the negotiation style:

<table>
<thead>
<tr>
<th>Negotiation Document</th>
<th>Supplier-side Response Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auction</td>
<td>Bid</td>
</tr>
<tr>
<td>RFI</td>
<td>Response</td>
</tr>
<tr>
<td>RFQ</td>
<td>Quote</td>
</tr>
</tbody>
</table>

When creating a negotiation style, you can create alternate labels for the category manager-side and/or supplier-side documents. Then any negotiation document created using that style will replace the default labels with the labels you created in the style. These replacements appear both in the online application and any printed versions of the document.

For example, you could use Tender or Offer for the supplier-side document, and you could use Solicitation for the buyer-side document.

Selecting Processing Capabilities and Defaults

In addition to specifying alternate document labels, you can select which processing capabilities are available using a negotiation style. By default, all capabilities are available. To create a style without that capability, simply deselect the capability when creating the style.

Manage Attribute Lists

Line Attribute Properties and Acceptable Values: Explained

Line attributes identify additional information about an item or service requested by the category manager. Use the following properties to describe your line attribute when defining it.

You use a line attribute’s properties to control its behavior and how the supplier should respond.
Response
You can control the supplier’s interaction with the attribute by specifying the response type. Optional responses do not require a supplier to offer a response. Suppliers must however provide a value for a required response. Suppliers can view attributes which are display only, but cannot respond to them.

Value Type
There are four value types available. A text value accepts characters and numbers. A number value accepts only numbers and the decimal point. Date values accept dates that you select using the calendar picker. A URL value type accepts a URL in the format http://url.name.here. URLs also accept https:

Target
For each attribute, you can define a target value. This is the value which is most desirable for this attribute. You can display the value to the supplier or keep it hidden.

Acceptable Values
For text values, you can specify a list of values from which the supplier can select. Any value not defined to the list is not accepted. For number and date values, you can define value ranges in terms of From Value and To Value. If you omit a From Value, that range includes everything up to the To value. Likewise, if you omit a To value, the range includes all values starting at the From value and above. Ranges cannot overlap. You can specify a single number by defining it as both the From and To values. Dates are defined similar to numbers.

Line Attributes and Multi Attribute Scoring Negotiations
In a multiattribute scored negotiation, you can have the application include the response values from the supplier along with the price offered when calculating the rank of that supplier’s response. To perform this calculation, you give each possible response value a numeric score and then weight the attribute among the other attributes for the line. Note that while each value’s score can be between 0 and 100, the weights for all the attributes must add up to 100.

Creating Attribute Lists: Worked Example

Attribute lists are collections of line attributes that are commonly used together. Procurement application administrators can create public attribute lists for use when creating negotiations. When a negotiation author associates an attribute list with a negotiation line, all the attributes on the list are associated with the line. Any attributes on the list that are not needed can be deleted, and additional line attributes can be created if necessary. Attribute lists are an efficient way to streamline the negotiation creation process. They can also be used to encourage standardization and best practices.

In this example, the procurement application administrator is going to create an attribute list.

Creating an Attribute List
1. On the Manage Attribute Lists page, she clicks the create icon.
2. When the Create Attribute List page appears, she enters the new list name.

3. She clicks the Actions menu option or the Add Group button to create an attribute group.

4. Once the new attribute group is created, she can create new attributes and save them as entries on the list.

Creating Line Attributes: Worked Example

Line attributes make your negotiation line more descriptive and can also be used to ensure that all responses submitted for the line include specific details not included elsewhere in the line information.

In this example, the category manager for a national used car seller is defining a negotiation to deal with suppliers of used cars. She defines the negotiation as a multiattribute scored auction and defines line attributes to describe vehicles.

Defining Line Attributes

1. After defining the negotiation line for the vehicle to buy, the category manager clicks the Edit icon at the end of the negotiation line for vehicle.

2. On the Lines: Edit Line page, she scrolls until the Attributes region appears. She clicks the Add Group button. On the new row, she clicks in the Attribute column and enters Vehicle Specifications for the group name. The other columns on the line are write-protected since they don’t apply to the Group element.

3. From the Actions menu, she selects the Add Attribute.

4. On the Add Attribute page, she enters Color as the Attribute. She accepts the defaults of Required for Response and Text for Value Type. She enters a weight of 20 for this attribute. She chooses to have no target value for this line attribute.

5. In the Acceptable Values region, she clicks the plus icon to add a row to the table. She proceeds to enter possible color values in the Response Value column and their numeric scores in the Score column. When finished, she clicks Save and Close to return to the Lines: Edit Line page.

6. She highlights the row for the Vehicle Specifications group and selects Add Attribute from the Actions menu to add another attribute to the negotiation line.

7. She names this attribute Mileage and this time selects Number as the Value Type. It has a weight of 80. She specifies a Target value of 5000 and displays this target to suppliers.

What's an attribute list?

A line attribute list is a collection of line attributes that you can apply to a negotiation line. When you apply an attribute list, all the line attributes on the list are associated with that line. Once applied to the line, you can modify
the line attributes if necessary. You can also delete any attributes that are not appropriate to the line.

Manage Cost Factors

Cost Factors and Cost Factor Lists: Explained

Cost factors allow you to identify and negotiate on additional costs related to a line.

You can use cost factors to obtain a more realistic idea of the total cost of an item or service by factoring in any additional costs beyond just price. Such costs could include additional costs such as consulting or training. Cost factors can be added to a negotiation line, to lines in negotiation templates, or to collections of cost factors (called cost factor lists). A negotiation line can have more than one cost factor (of any type) defined to it. There are three types of cost factors you can create.

Fixed Amount Cost Factors

Fixed amount cost factors are specified as a set value for the line, regardless of the quantity of units being asked for by the line.

Per-Unit Cost Factors

Per-unit cost factors are specified as a set value that is multiplied by the quantity of units being asked for by the line.

Percentage of Line Price Cost Factors

Percentage of line price cost factors are specified as a percentage. The percentage of line price is calculated by multiplying the unit price by the percentage of line price cost factor value.

Cost Factor Lists

Once you create cost factors for your additional costs, you can create lists of cost factors. Buyers can then apply these cost factor lists to negotiation lines to quickly identify the commonly occurring secondary costs that also need to be negotiated. Buyers can create their own personal cost factor lists using cost factors that have been defined to the system.

Creating Cost Factors: Worked Example

Cost factors identify charges associated with a negotiation line in addition to price. For example, these could include charges for shipping and handling, retooling, or import duties.

Cost factors are calculated in one of three ways: fixed amount per line, fixed amount per unit of line, and percentage of line price. You can use these pricing bases to reflect the nature of the extra cost. The table shows a possible example cost factor for each type.
Vision Corporation is expanding into a new branch of its items and services. Negotiations dealing with this new area will need to negotiate new transportation-related costs with suppliers, specifically with the new Chicago office. The procurement application administrator is going to define several new cost factors. Once the cost factors are defined and enabled, they become available for buyers to use individually or as members of a cost factor list.

Creating Cost Factors

1. On the Manage Cost Factors page, the procurement application administrator reviews the existing cost factors to ensure there isn’t already a cost factor that would be appropriate for the list. Not seeing one, she clicks the icon to add a new cost factor.

2. When the new line appears, she enters “Shipping” as the name, “General shipping charge” as the description, and enters “CHI_SHIP” as the unique code. She selects Fixed-amount for the pricing basis lets the status default to active.

3. She clicks the plus icon to add additional rows and enter the information for the Hazard materials charge and Import tax cost factors.

4. When finished, she clicks Save to save the new definitions.

Manage Cost Factor Lists

What’s a cost factor list?

A cost factor list is a collection of cost factors that you can apply to a negotiation line. When you apply a cost factor list, all the cost factors on the list are associated with that line. You can delete any cost factors you don’t need. Cost factor lists allow you to quickly associate a group of related or commonly used cost factors with a negotiation line. This speeds up the creation process.

Manage Negotiation Templates

Creating a Negotiation Template: Worked Example

If the majority of the negotiations you create contain the same features, for example, line attributes, terms and conditions, response controls; or negotiation data such as the value for the Location field, you may want to create a public
Define Sourcing Configuration

negotiation template that category managers can use each time they create a new negotiation (category managers can create their own private templates).

Using a template saves time by streamlining the creation process. A negotiation template contains the features that are similar among the negotiations you commonly create. When category managers create new negotiations using templates, they use the template as a shell for the negotiation, add to and edit details of the negotiation as necessary, and publish the negotiation.

In this example, the procurement application administrator will define a negotiation template to be used when negotiating for a list of incumbent supplies from Vision’s inventory category 200.13 which contains printers.

Creating a Negotiation Template

1. From the Manage Templates page, on she clicks the Create icon.

2. When the pop-up window appears, she selects the procurement business unit, negotiation type, a negotiation style if appropriate, negotiation outcome, and negotiation currency. Procurement business unit, negotiation outcome and negotiation currency are required fields. She then clicks Create.

3. She uses the train stops displayed across the top of the pages to create the negotiation template. This process is similar to the general negotiation creation process, but some fields and attributes are not available for use. For example, you cannot define a close date. The category manager will enter that date when she uses the template to create a negotiation.

4. She enters "Printer Auctions" as the template name, and on the Lines page, enters 200.13 for Category. On the Suppliers page, she enters the names of the suppliers with whom the company has purchased printers in the past.

5. When the procurement application administrator has completed the appropriate fields, she activates and saves the new template.

What’s the difference between a negotiation style and a negotiation template?

You can use both negotiation styles and negotiation templates when creating negotiation documents. The purpose for each is different, however, they both help shorten the creation process. A negotiation style uses only the creation features necessary to create the target type of negotiation. Any features not required are not accessible using that negotiation style. For example, if line attributes are not appropriate to a particular type of negotiation, the procurement application administrator can create a negotiation style that omits line attributes. When that negotiation style is used to create a negotiation document, the application pages used to create line attributes do not appear. By focusing only on the features required by the type of negotiation, the creation process is shortened.

A negotiation template is a skeleton you apply to a new negotiation you are creating. Negotiation templates can provide default data for many of the
negotiation document attributes, for example addresses or invited suppliers. You can modify some of these attributes when creating your new negotiation document.

Negotiation templates also let companies standardize practices on negotiation creation. For example, different templates can be used with different categories.
Define PROCUREMENT Contracts Configuration

Define Contract Terms Library Configuration

Contract Terms Library Setups: How They Work Together

This topic provides a brief overview of setups for the Contract Terms Library. The following figure outlines the main setups for the Contract Terms Library which are described in the sections of this topic. The setups on the left are accomplished using tasks from the Setup and Maintenance work area. To set up most of the Contract Terms Library features, including clauses and contract terms templates, you must navigate to the Terms Library work area. Dashed boxes highlight features that are available only in procurement contracts.
Setups in Oracle Fusion Functional Setup Manager

Different Oracle Fusion Functional Setup Manager tasks enable or affect Contract Terms Library features. These setups include:

- Setting Up Contract Types to Work with the Contract Terms Library
  
  Contract types specify properties of different contracts including the type of permitted contract lines, party roles, contract validation checks, and the contract acceptance and signature requirements. For the Contract Terms Library, you can use the Manage Contract Types task to:
  
  - Enable contract terms authoring
    
    You must enable contract terms authoring for a contract type to use any of the library features for contracts of that type.
  
  - Specify the Oracle BI Publisher layout template that will be used to format the printed contract terms for contracts of this type.

- Defining Clause Types
  
  If you want to categorize the clauses in the library, select the Manage Contract Clause Types task to set up clause types.

- Configuring Business Units for Contracts
  
  The use of most of the Contract Terms Library content is restricted to the business unit where you create it. This includes clauses, contract terms templates, and Contract Expert business rules. Using either the Specify Customer Contract Business Function Properties or the Specify Supplier Contract Business Function Properties tasks, you can:
  
  - Enable content adoption between business units and automatic approvals for content
  
  - Specify the Contract Terms Library administrator, the employee who will receive approvals and other notifications regarding library content.
  
  - Enable the Contract Expert feature for the business unit.

- Creating Contract Layout Templates
  
  Using Oracle BI Publisher, you can set up layout templates that determine the formatting of clauses, contract terms template previews, the contract deviations report, and the contract itself.

  Download the sample layout templates provided with your application from the Oracle BI Publisher library. You can copy and edit the sample layout templates and upload them.

- Creating Contract Terms Value Sets
  
  Select the Manage Contract Terms Value Sets task to set up value sets for use in contract terms variables and Contract Expert questions.

- Specifying the Location of the File Used for Clause Import
You can import legacy clauses into the Contract Terms Library, either from a file or from an interface table using Oracle Fusion Enterprise Scheduler processes.

If you are importing clauses from a file, then you must specify the location of the file by setting the system profile option **Specify Contract Clause Import XML File Location** by selecting the **Manage Clause and Template Management Profiles** task.

**Contract Terms Library Work Area Setups**

The Contract Terms Library is built using the tasks within the Terms Library work area:

- **Creating Clauses**

  Create standard clauses for use during contract terms authoring, including alternate clauses, clauses included by reference, and provision clauses. By specifying different clause properties, you can modify clause behavior. For example, you can make clauses mandatory in contracts or protect them from editing by contract authors.

- **Creating Variables**

  You can use variables in the Contract Terms Library to represent information within individual clauses and for use within Contract Expert rule conditions. Your application comes with predefined variables, called system variables. You can create additional variables, called user variables, with or without programming.

- **Creating Numbering Schemes**

  You can set up additional clause and section numbering for contract terms. You can select which numbering scheme you want to use with each contract terms template.

- **Creating Contract Terms Templates**

  Create contract terms templates to insert boilerplate terms and conditions into contracts during contract authoring. Contract authors can apply the templates manually, or the application can apply the templates automatically using defaulting rules you set up.

- **Creating Contract Expert Business Rules**

  Set up business rules that ensure compliance of contracts with corporate standards.

  Contract Expert makes it possible for you to set up business rules that can:

  - Apply the appropriate contract terms template to a contract

    For example, apply the contract terms template Software License and Service Agreement if the contract is authored in the North America Operations business unit and the contract amount exceeds one million dollars.
• Insert additional clauses into specific predetermined locations in the contract
  For example, add an audit clause if an audit is required.
• Report contract deviations from corporate policies
  For example, report a contract worth one million dollars or more that includes payment terms greater than 90 days.

You can base Contract Expert rule conditions on the values of variables in the contract, the presence of other clauses, or you can set up questions that contract authors must answer during authoring.

For example, you can ask authors a series of questions about the nature of the materials being shipped to customers and insert additional liability clauses based on their answers.

If you are setting up business rules with numeric conditions (for instance, insert a special payment terms clause if the contract amount exceeds $1 million) then you must set up constants to hold the numeric values. You cannot enter the numeric values directly.

• Contract Deliverables

Contract deliverables track both contractual and non-contractual commitments that must be completed as part of negotiations, purchasing, and enterprise contracts between businesses and suppliers or customers based on contract intent. These deliverables can be used in purchasing and sourcing documents that include contract terms and in enterprise contracts.

• Importing Clauses

You can import clauses from legacy applications by running Oracle Fusion Enterprise Scheduler (ESS) processes from the Terms Library work area by selecting the Import Clauses task or from the Setup Manager by selecting the Manage Processes task.

• Setting Up and Maintaining the Index for Clause Text Searches Using the Keyword Field

By selecting the Manage Processes task in the Terms Library work area, you can also run the ESS processes required to set up and maintain the text index required for searches of clauses and contract terms templates using the Keyword field.

**Contract Terms Library Clauses: Explained**

You can create different types of clauses for different uses and use clause properties to specify if a clause is protected from edits by contract authors, if it is mandatory, and if it is related to or incompatible with other clauses. A clause you create in the Contract Terms Library is available for use within the business unit where you create it after it is approved.
The types of clauses you can create include:

- Standard clauses
- Clauses included by reference
- Provision clauses for contracts with a buy intent

Using different clause properties you can:

- Make a clause mandatory in a contract.
- Protect it from edits by contract authors.
- Specify that a clause can be selected by contract authors as an alternate of another clause.
- Specify that the clause cannot be in the same document as another clause.
- Make a clause created in a global business unit available for use in other business units.

**Creating Standard Clauses**

Any clause you create in the library becomes a standard clause that can be used in the business unit where you create it after it is approved. Unless you specify that the clause is protected, contract authors can edit the clause in a specific contract. Any edits they make are highlighted in a clause deviations report when the contract is approved. Similarly, contract authors can delete the clause from a contract, unless you specify the clause is mandatory.

**Including Clauses by Reference**

For clauses, such as Federal Acquisition Regulation (FAR), you can print the clause reference in the contract instead of the clause text itself. During contract creation, you enter the reference on the Instructions tab of the clause edit page and select the **Include by Reference** option.

**Creating Provision Clauses for Contracts with a Buy Intent**

For contracts with a buy intent, you can create provision clauses, clauses that are included in contract negotiations but are removed after the contract is signed. Provision clauses are used primarily in Federal Government contracting.

**Altering Clause Behavior with Clause Properties**

Using different clause properties, you can alter the behavior of a clause. You can:

- Make a clause mandatory.

  A mandatory clause is highlighted by a special icon during contract terms authoring and cannot be deleted by contract authors without a special privilege. You can make a clause mandatory for a particular contract terms template by selecting the Make Mandatory action after you have
added the clause to the template. A clause is also become mandatory if it is added by a Contract Expert rule and you have selected the Expert Clauses Mandatory option in the template.

- Protect it from edits by contract authors.

A protected clause is highlighted by a special icon during contract terms authoring and cannot be edited by contract authors without a special privilege. You can protect any clause by selecting the protected option during clause creation or editing.

- Specify that a clause can be selected by contract authors as an alternate of another clause.

You can specify clauses to be alternates of each other on the Relationships tab of the create and edit clause pages. When editing contract terms, contract authors are alerted by an icon that a particular clause includes alternates and can select an alternate to replace the original clause.

- Specify that the clause cannot be in the same document as another clause.

You can use the Relationship tab to specify a clause you are creating is incompatible with another clause in the library. The application highlights incompatible clauses added by contract authors in the contract deviations report and during contract validation.

- Make a clause available for use in other business units.

Clauses you create in the library are normally available only within the same business unit where you create them. If you create the clause in the business unit that is specified as global during business unit setup, then you can make the clause available for adoption in other business units by selecting the Global option during clause creation or edit. This option appears only in the one business unit specified as global.

**Contract Terms Templates: How They Work**

You can create contract terms templates in the Contract Terms Library to insert appropriate terms and conditions into contracts during contract authoring. Contract authors can apply the templates manually or the application can apply the templates automatically using defaulting rules you set up.

Contract terms templates:

- Contain sections and clauses from the Contract Terms Library.

- Are created in the Contract Terms Library separately. You cannot create them directly from an existing contract.

- Are specific to one business unit.

- Apply to enterprise contracts of the contract types you specify in the template.

- Are specific to either sell-intent or buy-intent contracts.
• Can default contract terms directly on purchase orders and sourcing documents, and on enterprise contracts. For these documents, contact terms templates can also include contract deliverables which can be used to track the completion of contractual tasks in the contract.

In addition, for a contract terms template you can:

• Set up Contract Expert rules to recommend additional clauses for contracts that use the template and insert these clauses in specified locations in the contract if marked as conditional.

• Associate a layout template for previewing the template.

• Specify a contract terms numbering scheme for the template.

• Set up template selection rules to default the template into a contract automatically.

Adding Sections
You can add sections that you have created in the library or create sections that are specific to the template itself.

Adding Clauses
You can add clauses in one of two ways:
• Add a clause from the Contract Terms Library directly into a section in the template.

You can create the clause in the library from the template if the library does not have what you need.

• Create Contract Expert rules to add clauses to the contract terms in a contract depending on the specifics of the contract.

For example, you may want to add a boilerplate jurisdiction clause directly into the template, but use a Contract Expert rule to insert the appropriate liability clause. This way, a contract that calls for the shipment of hazardous materials will get a liability clause that’s different from a contract that does not include any, for example.

The properties that you set up in the clause apply automatically. If you set up a clause as mandatory, contract authors will not be able to delete the clause after it is inserted by the template unless they have the special Override Contract Terms and Conditions Controls privilege. If you set up a clause with alternates, then authors can substitute any of the alternate clauses in the contract.

Note
You are not required to add any sections or clauses to a template directly. You can use Contract Expert rules exclusively, if appropriate.

Enabling Contract Expert on the Template

If you want to use Contract Expert in a contract where the template is applied, you must enable the template for Contract Expert by selecting the Enable option in the Contract Expert region in the Create Terms Template or Edit Terms Template pages.

When Contract Expert rules enabled for the template suggest additional clauses, these additional clauses are presented for review by contract authors before they are inserted in the default section specified in each clause. Depending on their level of privileges, some contract authors can choose which clauses to insert and which to omit. If you make Contract Expert suggestions mandatory for the template, then only users with the special Override Contract Terms and Conditions Controls privilege can reject the recommendations.

The clauses recommended for insertion may also be placed in their predetermined locations, if the clauses are marked as conditional clauses and their locations defined in the terms template associated with the contract.

Adding Contract Deliverables to Purchase Orders, Sourcing Documents, and Enterprise Contracts

For Oracle Fusion Purchasing purchase orders, Oracle Fusion Sourcing documents, and enterprise contracts, you can track compliance of tasks that the contract parties have agreed to execute as part of the agreement by adding contract deliverables.

You can use the deliverables to record the status of the tasks, keep everyone notified of past and future deadlines, and as a repository of the deliverable documents themselves. For example, vendors agreeing to supply a monthly
report can log in to their sourcing portal and attach the report or ask for an extension. If they fail to respond by the specified deadline, the deliverable can trigger an automatic notification that the deliverable is overdue.

**Assigning a Layout Template for Previewing the Contract Terms Template**

You must assign a layout template with the contract terms template to make it possible for contract authors to get a preview of the template content, when they need to make a template selection, for example. The layout template, which you select on the General tab while editing the contract terms template, specifies what gets displayed in the preview, including the fields displayed, graphics such as a company logo, page numbering, headers and footers, and boilerplate text. This layout template is not used for printing the contract.

If you marked Contract Expert recommended clauses as conditional on the terms template, then these are displayed in gray font in the print preview to distinguish them from regular clauses.

The layout template is an RTF file stored in the Enterprise Contracts folder in the Business Intelligence Presentation Catalog. A sample layout template is provided with your application. You can copy the sample template and edit it to create your own as described in a related topic.

**Specifying a Numbering Scheme**

You can associate a numbering scheme to the template that will automatically number sections and clauses in the contract. Several predefined numbering schemes are available with your application, and you can create additional numbering schemes of your own.

**Defaulting the Template on Contracts**

You can have a contract terms template apply automatically in all contracts based on:

- Contract type
- Contract Expert rules that select the template based on the specific information in the contract itself

If you enabled the feature Enable Contract Terms in Fusion Procurement for the option Procurement Contracts during implementation, then you can also apply templates to procurement documents based on document type.

The following document types become available:

- Auction
- Bid
- Blanket Purchase Agreement
- Contract Purchase Agreement
- Standard Purchase Order
- RFI
- RFI Response
• RFQ
• Sourcing Quote

While editing the contract terms template, you specify a template to be the default for a contract type or document type in the Document Types region. You can set up only one template as the default for each contract type or document type. You set up the Contract Expert template selection rules separately as described in a related topic. You can have multiple rules recommend the same template.

Here is how the defaults you enter in the Document Types region and the Contract Expert template selection rules interact to select and apply a template during contract authoring:

• Contract Expert template selection rules always take priority. If the rules specify a single template for a contract, then it gets applied regardless of the default you entered in the Document Type region.

• If the Contract Expert rules recommend different templates, then the application uses the default from the Document Type region as a tiebreaker.

• If no Contract Expert selection rule applies and you specified a default, then the application uses the default.

• If no rule or default was set up for a contact type or document type, then contract authors must select the template they want from a list.

**Contract Expert Rules: How They Work**

You can set up Contract Expert rules to apply contract terms templates automatically to contracts, to suggest additional clauses for insertion during contract terms authoring, and to flag any contract deviations from company policy.

Each rule comprises conditions that must be met and the rule results. You can base rule conditions on:

• The presence of another clause already in the contract
• The value of a system variable or a user variable
• Questions that the contract author must answer

Different Contract Expert rule types support different condition types, as illustrated in the following figure.

• Clause selection rules, which can default individual clauses and sections into a contract, can be based on clauses, questions, and variables.

• Template selection rules, which identify the default contract terms template for the contract, can be based on variables only.

• Policy deviation rules, which flag contract deviations from company policies, use questions and variables only.
Key rule properties include:

- All rules can use multiple conditions linked together with either the AND or OR logical operators.
- The values of non-numeric conditions are supplied by value sets.
- The values for numeric conditions are supplied by constants.
- Rule types that permit the inclusion of questions can trigger follow-up questions, permitting you to chain rules together.
- Rules are restricted to the specific business unit and the contract intent where you create them.
- Rules do not get copied when you copy a global contract terms template to another business unit.
- Conditions support both logical and numeric operators:
  - IS
  - IS NOT
  - IN (allows the selection of multiple values)
  - NOT IN (allows the selection of multiple values)
  - >=: (greater than or equal to)
  - <=: (less than or equal to)
  - := (equal to)
  - > (greater than)
Clause Selection Rules

Clause selection rules permit you to insert one or more clauses and sections into a contract.

The following table describes the rule properties.

<table>
<thead>
<tr>
<th>Rule Property</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule outcomes</td>
<td>The rule can:</td>
</tr>
<tr>
<td></td>
<td>• Recommend one or more clauses for insertion into the contract</td>
</tr>
<tr>
<td></td>
<td>Contract authors can review the Contract Expert recommendations before the clauses get inserted into the contract. By setting the <strong>Expert Clauses Mandatory</strong> option when creating a contract terms template, you can specify if you want the clause insertion to be mandatory or if the authors can ignore the recommendations. If you make the insertion mandatory, then only contract authors with the Override Contract Terms and Conditions Controls privilege, a special privilege that allows deleting mandatory clauses from the contract, can reject the recommendations. Similarly, if the recommended clauses are standard clauses, then the authors must have the <strong>Author Additional Standard Contract Terms and Conditions</strong> privilege to reject the recommendations. This privilege allows the deletion of standard clauses from the contract. If you marked recommended clauses as conditional and specified the location of these clauses in the terms template, then Contract Expert inserts the clause in the contract in the location that you specified. If the location of an Expert suggested clause is not specified in the terms template, Contract Expert inserts each clause in the section specified as the default for the clause in the Contract Terms Library. If no default section is specified in the clause, then Contract Expert inserts the clause into the default section specified in the contract terms template. Contract Expert automatically inserts the default section if it does not already exist in the contract.</td>
</tr>
<tr>
<td></td>
<td>• Ask follow-up questions</td>
</tr>
<tr>
<td></td>
<td>You can ask follow-up questions by adding them in the Additional Questions region of the Results tab. Any additional question that you add must be part of another rule. Adding the follow-up question chains the rules together.</td>
</tr>
</tbody>
</table>
When the rule is evaluated

<table>
<thead>
<tr>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>The rule is evaluated every time that a user runs Contract Expert.</td>
</tr>
<tr>
<td>Users receive an warning message during contract validation if they fail to run Contract Expert.</td>
</tr>
</tbody>
</table>

Conditions

<table>
<thead>
<tr>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conditions can be based on:</td>
</tr>
<tr>
<td>• clauses</td>
</tr>
<tr>
<td>• questions</td>
</tr>
<tr>
<td>• variables</td>
</tr>
<tr>
<td>You can use both predefined system variables and user variables. Both types of user variables are supported: those that require entry by contract authors and those where the values are supplied by a Java procedure.</td>
</tr>
</tbody>
</table>

Where it applies

<table>
<thead>
<tr>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>The rule applies only within the business unit and for the intent that you specify. You can have the rule apply to one of the following:</td>
</tr>
<tr>
<td>• Specific contract terms templates</td>
</tr>
<tr>
<td>• All contract terms templates for the business unit</td>
</tr>
</tbody>
</table>

**Contract Terms Template Selection Rules**

Contract terms template selection rules permit you to automatically apply a contract terms template to a contract.

The following table describes the rule properties.

<table>
<thead>
<tr>
<th>Rule Property</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule outcomes</td>
<td>The application automatically applies a contract terms template to a contract. Or, if the author removed the contract terms using the Actions menu, the template displays the template name as the default when applying a new template.</td>
</tr>
<tr>
<td>When the rule is evaluated</td>
<td>The application evaluates the rule whenever the author navigates to the Contract Terms tab as long as no contract terms template is applied. If a contract terms template is applied to the contract, the template selection rules are not executed again, even if changes to the contract would result in a different rule outcome. The rule is also evaluated to determine if the contract contains the recommended template whenever the contract author:</td>
</tr>
<tr>
<td></td>
<td>• runs the clause deviations report</td>
</tr>
<tr>
<td></td>
<td>• validates the contract terms or the contract</td>
</tr>
<tr>
<td></td>
<td>In both cases, the rule generates a warning if the author applied a different template from that recommended by the rule.</td>
</tr>
</tbody>
</table>
Contract Expert does not apply a contract terms template if the contract terms template defaulting rules you set up recommend multiple terms templates for a single contract. Instead, Contract Expert applies the contract terms template specified as the default for the business document type during contract terms template setup. If no document type default is specified, then the application displays the **Add Contract Terms** button and permits authors to select a template of their own choice. The choices are restricted to the templates specified for the contract type.

Policy Deviation Rules

Policy deviation rules flag deviations from company policies on the contract deviations report. This report is run by the contract author before submitting a contract for approval.

The following table lists the rule properties.
<table>
<thead>
<tr>
<th>Rule Property</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule results</td>
<td>The rule displays a deviation in the contract deviations report. The rule name becomes the deviation.</td>
</tr>
</tbody>
</table>
| When the rule is evaluated | The rule is evaluated whenever the user:  
  - Runs the contract deviations report  
  - Validates the contract terms or the contract                                                                                                   |
| Conditions         | Conditions can based on:  
  - Questions  
  - Variables  
    Both predefined system variables and those user-defined variables where the values are supplied by a Java procedure.                                                                               |
| Where it applies   | The rule applies only for the contract terms templates within the business unit and for the intent that you specify.                                                                                           |

The following figure illustrates the policy deviation rule setup. You can build rule conditions out of both questions and variables. In the contract deviation report, your entry in the **Rule Name** field becomes the deviation name and your entry in the rule **Description** field becomes the deviation description.

Policy deviation rules list policy deviations in the contract deviations report, along with any clause deviations that are flagged automatically by the application. Contract authors can run the report before submitting the contract for approval and enter comments to explain the deviation to the approver. The report is rerun automatically when the author submits the contract for approval and a copy of the report is attached to the approval notification.
Activating and Validating Rules

After you set up a rule, you must activate it using the Activate Rule action. Rules do not require approval before activation, but the contract terms templates that they apply to do.

Note

In order to activate a rule, you must assign it to at least one contract terms template. The template does not have to be approved at the time that you make the assignment, but it does have to be approved before the rule can be used.

Activating a rule triggers an automatic validation process. You must correct all errors before the rule gets activated.
Web Services: Overview

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 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>
</table>
| ADF services     | 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. Examples of ADF services include:  
  * Worker.changeHireDate - a service that updates the hire date of the worker business object.  
  * ProjectTask.createTask - a service that adds a task to the project task business object. |
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. For information or assistance regarding general access to content management (including all metadata), to create and manage accounts, and to programatically upload and download content, contact the WebCenter Content Administrator.
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. 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
- Searching records
- Accessing content in a new account
- Account names
- Deleting files

**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 $. For example fin$/journal$/import$ displays as fin/journal/import. The Remote Introdoc 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><strong>FusionApps: Logical Business Area</strong></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 want to 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.
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>fin/receivables/import</td>
</tr>
<tr>
<td>• Customer Import</td>
<td>fin/cashManagement/import</td>
</tr>
<tr>
<td>• China Value Added Tax Invoice Import</td>
<td>fin/cashManagement/import</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>fin/cashManagement/import</td>
</tr>
<tr>
<td>• ISO200022 CAMT053 Format Bank Statements Import</td>
<td>fin/cashManagement/import</td>
</tr>
<tr>
<td>• SWIFT MT940 Format Bank Statements Import</td>
<td>fin/cashManagement/import</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>fin/assets/import</td>
</tr>
<tr>
<td>• Fixed Asset Mass Retirements Import</td>
<td>fin/assets/import</td>
</tr>
<tr>
<td>• Fixed Asset Mass Transfers Import</td>
<td>fin/assets/import</td>
</tr>
<tr>
<td>• Fixed Asset Units of Production Import</td>
<td>fin/assets/import</td>
</tr>
<tr>
<td>Intercompany Transaction Import</td>
<td>fin/intercompany/import</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------</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>fin/generalLedger/import</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>
<tr>
<td>Import Purchase Orders</td>
<td>prc/purchaseOrder/import</td>
</tr>
<tr>
<td>Import Requisitions</td>
<td>prc/requisition/import</td>
</tr>
<tr>
<td>• Import Suppliers</td>
<td>prc/supplier/import</td>
</tr>
<tr>
<td>• Import Supplier Sites</td>
<td>prc/supplier/import</td>
</tr>
<tr>
<td>• Import Supplier Site Contacts</td>
<td>prc/supplier/import</td>
</tr>
<tr>
<td>• Import Supplier Site Assignments</td>
<td>prc/supplier/import</td>
</tr>
<tr>
<td>Project Enterprise Resource Import</td>
<td>prj/projectManagement/import</td>
</tr>
<tr>
<td>Project Unprocessed Expenditure Item Import</td>
<td>prj/projectCosting/import</td>
</tr>
<tr>
<td>Cycle Count Import</td>
<td>scm/cycleCount/import</td>
</tr>
<tr>
<td>Inventory Reservation Import</td>
<td>scm/inventoryReservation/import</td>
</tr>
<tr>
<td>Inventory Transaction Import</td>
<td>scm/inventoryTransaction/import</td>
</tr>
<tr>
<td>Item Import</td>
<td>scm/item/import</td>
</tr>
<tr>
<td>Receiving Receipt Import</td>
<td>scm/receivingReceipt/import</td>
</tr>
<tr>
<td>Shipment Request Import</td>
<td>scm/shipmentRequest/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.

**Load Interface File for Import Process**

Use to load external setup or transaction data from a data file in the content repository to interface tables. The process prepares the data for import into application 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.
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. &lt;br&gt;moduleType/moduleKey: only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. &lt;br&gt;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. &lt;br&gt;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 Category | 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. |
<p>| 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. |</p>
<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. moduleType/moduleKey: only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. lookupType: only the specified common lookup is moved.</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. moduleType/moduleKey - only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. lookupType: only the specified common lookup is moved.</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. moduleType/moduleKey: only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. lookupType: only the specified set-enabled lookup is moved.</td>
</tr>
<tr>
<td>Application Profile Category</td>
<td>Profile categories</td>
<td>No parameters: all profile categories are moved. moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. name/applicationId: only the specified category is moved.</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  | Reference data set assignments  | determinantType: only assignments for the specified determinant type are moved.  
| Assignment                      |                                 | 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. |
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.

accounting method
A set of journal entry rules which determine how a subledger journal entry is to be created for each event class or event type.

action
The kind of access named in a security policy, such as view or edit.

activity stream
A feature that tracks and displays actions and messages from people whom you are connected to in your social network, as well as activities from the application.

ADF
Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

application feature
A standardized functionality that is available to implemented.

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.

**automatic assignment catalog**

A non-hierarchical catalog to which categories that match the catalog’s Catalog Structure value are automatically added. Add categories and share categories actions are disabled for this catalog configuration.

**balancing segment**

A chart of accounts segment used to automatically balance all journal entries for each value of this segment.

**beneficiary**

A person or organization designated to receive benefits from a compensation plan on the death of the plan participant.

**BPEL**

Business Process Execution Language; a standard language for defining how to send XML messages to remote services, manipulate XML data structures, receive XML messages asynchronously from remote services, manage events and exceptions, define parallel sequences of execution, and undo parts of processes when exceptions occur.

**browsing category**

Parent or intermediate category that is associated with other categories in the catalog hierarchy, but has no assigned items.

**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.
**catalog**
A collection of categories used to classify items which can be organized into a hierarchy that represents a taxonomy.

**category**
Catalog component that is associated to a catalog to classify items.

**chart of accounts**
The account structure your organization uses to record transactions and maintain account balances.

**clause adoption**
Reusing a clause from the global business unit in local business units either by adopting the clause without change or by localizing it.

**clause localization**
A type of clause adoption where the adopted clause is edited to suit the local business unit needs.

**condition**
An XML filter or SQL predicate WHERE clause in a data security policy that specifies what portions of a database resource are secured.

**constant**
Holds the numeric value used to evaluate numeric conditions in Contract Expert rules. A constant permits you to reset the conditions of many rules with just one edit.

**context**
A grouping of flexfield segments to store related information.

**context segment**
The flexfield segment used to store the context value. Each context value can have a different set of context-sensitive segments.

**context-sensitive segment**
A flexfield segment that may or may not appear depending upon a context such as other information that has been captured. Context-sensitive segments are custom attributes that apply to certain entity rows based on the value of the context segment.

**contingent worker**
A self-employed or agency-supplied worker. Contingent worker work relationships with legal employers are typically of a specified duration. Any
person who has a contingent worker work relationship with a legal employer is a contingent worker.

**contract deliverable**

A task that needs to be performed as part of the execution of a contract or business document, and that is tracked as part of the contract terms and conditions.

**contract deviations**

Differences between the contract terms in a contract and those in the contract terms template applied to that contract and any deviations from company policies as determined by Contract Expert feature rules.

**Contract Expert**

A feature of the application that permits you to create business rules in the Contract Terms Library to enforce corporate policies and standards for contracts.

**Contract Terms Library**

A repository of standard clauses, contract terms templates, and business rules maintained by your organization.

A repository of standard clauses, contract terms templates, and business rules built using Contract Expert.

**contract terms template**

A template of standard clauses set up in the Contract Terms Library applied during contract authoring either automatically by the application or manually by contract authors.

**contract type**

A setup that specifies enterprise contract content, including the presence of contract terms and contract lines.

**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 Factor

Cost factors allow a buyer to identify and control for additional costs associated with a negotiation line. Cost factors can be calculated as either a per-unit cost, a percentage of the line price, or a fixed amount for the line.

cost organization

A grouping of inventory organizations that indicates legal and financial ownership of inventory, and which establishes common costing and accounting policies.

country holding company

A legal entity that acts on behalf of several divisions within an enterprise, and is the legal employer in a country.

data dimension

A stripe of data accessed by a data role, such as the data controlled by a business unit.

data instance set

The set of human capital management (HCM) data, such as one or more persons, organizations, or payrolls, identified by an HCM security profile.

data role

A role for a defined set of data describing the job a user does within that defined set of data. A data role inherits job or abstract roles and grants entitlement to access data within a specific dimension of data based on data security policies. A type of enterprise role.

data role template

A template used to generate data roles by specifying which base roles to combine with which dimension values for a set of data security policies.

data security

The control of access to data. Data security controls what action a user can taken against which data.

data security policy

A grant of entitlement to a role on an object or attribute group for a given condition.

database resource

An applications data object at the instance, instance set, or global level, which is secured by data security policies.
department
A division of a business enterprise dealing with a particular area of activity.

descriptive flexfield
Customizable expansion space, such as fields used to capture additional descriptive information or attributes about an entity, such as customer cases. Information collection and storage may be configured to vary based on conditions or context.

desktop page
A page in a user interface that is optimized for extended periods of use with monitors.

determinant
A value that determines which reference data set will be used in a specific business context.

determinant type
Designates the field within transactional columns that controls how data is shared across organizations such as business unit, asset book, cost organization or project unit. The type determines the reference data sets that would be used in a transaction.

determinant type
An additional and optional field within transactional columns (besides category and application) that is used to assign document sequences. The available determinant types are Business Unit, Ledger, Legal Entity, and Tax Registration.

determinant value
A value specific to the determinant type dimension of a document sequence. The determinant value is relevant in a document sequence assignment only if the document sequence has a determinant type. If Ledger is the determinant type for a document sequence, the determinant value is the specific ledger number whose documents are numbered by the document sequence.

division
A business-oriented subdivision within an enterprise. Each division is organized to deliver products and services or address different markets.

document sequence
A unique number that is automatically or manually assigned to a created and saved document.

document type
A categorization of person documents that provides a set of options to control what document information to retain, who can access the documents, whether
the documents require approval, and whether the documents are subject to expiration. A document type exists for a combination of document category and subcategory.

**document type**

A categorization of contracts, including auction, blanket purchase agreement, contract purchase agreement, RFI, RFQ, standard purchase order, and enterprise contract.

**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.

**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 contract**

A contract created in the Oracle Fusion Enterprise Contracts application.

**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.

**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 let you group 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.

gallery
A searchable collection of portraits that combines the functions of the person directory with corporate social networking and self-service applications for both workers and managers.

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.

global business unit
A business unit, designated as global during business unit setup, that can make its clauses and contract terms templates available for adoption by local business units.

grade
A component of the employment model that defines the level of compensation for a worker.

HCM
Abbreviation for Human Capital Management.

HCM data role
A job role, such as benefits administrator, associated with specified instances of Oracle Fusion Human Capital Management (HCM) data, such as one or more positions or all persons in a department.
**HCM securing object**

An HCM object that secures access to both its own data and data in other, related objects. For example, access to a specified set of person records can allow access to data secured by person records, such as goal plans and evaluations.

**identity**

A person representing a worker, supplier, or customer.

**import**

In the context of data integration, the transfer of data from interface tables to **application tables**, where the data is available to application users.

**intent**

Specifies if an object in the Contract Terms Library is used for procurement contracts or for sales contracts.

**interface table**

A database table used for transferring data between applications or from an external application or data file.

**inventory organization**

A logical or physical entity in the enterprise that is used to store definitions of items or store and transact items.

**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.

**item subinventory**

An association of an item with a subinventory that is created when you add an item to a subinventory.

**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.

legal authority
A government or legal body that is charged with powers such as make laws, levy and collect fees and taxes, and remit financial appropriations for a given jurisdiction.

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 jurisdiction
A physical territory, such as a group of countries, single country, state, county, parish, or city, which comes under the purview of a legal 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 Attribute**

When you add a line to a negotiation, you may choose to add one or more line attributes to that line. Line attributes define unique specifications that you set for a negotiation line and the details that a supplier should provide when responding to that negotiation line. Line attributes can be used to ensure that all responses submitted for the line include important details beyond just the price offered for the line.

**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.

**load**

In the context of data integration, the transfer of external data from data files to the receiving interface tables in preparation for an import into application tables.

**lookup code**

A value available for lookup within a lookup type such as the code BLUE within the lookup type COLORS.

**lookup type**

A set of lookup codes to be used together as a list of values on a field in the user interface.

**mainline**

A branch of data that serves as a single source of truth.

**managed person**

In Oracle Fusion Human Capital Management security, a person for whom the user can maintain some information. For example, line managers can maintain information about their direct and indirect reports, and workers can maintain information about themselves, their dependents, and their beneficiaries.

**manufacturing facilities**

Employed in the making of goods for sale such as a factory or plant.

**model profile**

A collection of the work requirements and required skills and qualifications of a workforce structure, such as a job or position.
native catalog
A catalog that a user is managing.

natural account
Categorizes account segment values by account type, asset, liability, expense, revenue, or equity, and sets posting, budgeting, and other options.

natural account segment
A chart of accounts segment used to categorize your accounting transactions by account type: asset, liability, owner's equity, revenue, or expense.

numbering scheme
The style of numbering used for the sections and clauses in contract terms.

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.

OWLCS
Abbreviation for Oracle WebLogic Communication Services. Offers the TPCC service to Oracle Sales Cloud and sets up the calls via SIP integration with the telephony network.

party
A physical entity, such as a person, organization or group, that the deploying company has an interest in tracking.

party fiscal classification
A classification used by a tax authority to categorize a party for a tax.

payroll statutory unit
A legal entity registered to report payroll tax and social insurance. A legal employer can also be a payroll statutory unit, but a payroll statutory unit can represent multiple legal employers.

pending worker
A person who will be hired or start a contingent worker placement and for whom you create a person record that is effective before the hire or start date.

person 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.
person type
A subcategory of a system person type, which the enterprise can define. Person type is specified for a person at the employment-terms or assignment level.

personally identifiable information
Any piece of information that can potentially be used to uniquely identify, contact, or locate a single person. Within the context of an enterprise, some PII data can be considered public, such as a person's name and work phone number, while other PII data is confidential, such as national identifier or passport number.

PL/SQL
Abbreviation for procedural structured queried language.

portrait
A selection of information about a worker or nonworker, including contact details, social connections, and activities and interests, that can be viewed and edited. Both the amount and type of information and the available actions depend on the role of the portrait user.

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 balancing segment value
A segment value used to represent a legal entity in the chart of accounts and automatically balance all intercompany and intracompany transactions and journal entries.

primary ledger
Main record-keeping ledger.

privilege
A grant or entitlement of access to functions and data. A privilege is a single, real world action on a single business object.

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.

**provision clause**
A clause that is used only in negotiations and is dropped when the negotiation is converted to a contract.

**PSTN**
Abbreviation for public switched telephone network which is the network of the world’s public circuit-switched telephone networks.

**public person**
In Oracle Fusion Human Capital Management security, a person for whom some basic information is publicly available. For example, users typically access the contact details of public persons, such as phone numbers and locations, using the person gallery.

**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.

**referenced category**
A category within the native catalog that is shared from a designated source catalog. A reference category is not editable.
**registration**

The record of a party's identity related details with the appropriate government or legal authorities for the purpose of claiming and ensuring legal and or commercial rights and responsibilities.

**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.

**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 run time session that commits changes out of reach of mainline users.

**security profile**

A set of criteria that identifies one or more human capital management (HCM) objects of a single type for the purposes of securing access to those objects. Security profiles can be defined for persons, organizations, positions, countries, LDGs, document types, payrolls, and payroll flows.

**security reference implementation**

Predefined function and data security in Oracle Fusion Applications, including role based access control, and policies that protect functions, data, and segregation of duties. The reference implementation supports identity management, access provisioning, and security enforcement across the tools, data transformations, access methods, and the information life cycle of an enterprise.
segregation of duties
An internal control to prevent a single individual from performing two or more phases of a business transaction or operation that could result in fraud.

service provider model
A business unit that provides specific business functions for another business unit.

set
Reference data that is organized into groups appropriate to organizational entities, to enable reference data sharing.

set enabled
An entity, such as a lookup, customer, location, organization, or document attachment, that is allowed to participate in reference data sharing by drawing on the data of a reference data set.

shared category
A category within a source catalog that has been added to a native catalog as a referenced category. The category can be shared with one or more catalogs.

simplified page
A page in a user interface that is optimized to provide quick access to high-volume, self-service tasks from any device.

SOA
Abbreviation for service-oriented architecture.

spot rate type
Rate you enter to perform conversion based on this rate as of a specific date. This rate applies to the immediate delivery of a currency.

SQL predicate
A type of condition using SQL to constrain the data secured by a data security policy.

storage facilities
Commercial building for storage of goods such as a warehouse.

system person type
A fixed name that the application uses to identify a group of people.
**system variable**
A predefined variable that gets its value from an attribute of the contract or other document.

**tax**
The classification of a charge imposed by a government through a fiscal or tax authority.

**tax exemption**
A full or partial exclusion from taxes within a given time period.

**tax jurisdiction**
A geographic area where a tax is levied by a specific tax authority.

**tax rate**
The rate specified for a tax status for an effective time period. A tax rate can be expressed as a percentage or a value per unit quantity.

**tax recovery**
The full or partial reclaim of taxes paid on the purchase or movement of a product.

**tax regime**
The set of tax rules that determines the treatment of one or more taxes administered by a tax authority.

**tax registration**
The registration of a party with a tax authority that confers tax rights and imposes certain tax obligations.

**tax rule**
A user-defined rule that looks for a result for a specific tax determination process, such as determining place of supply or tax registration, in relation to a tax on a transaction.

**tax status**
The taxable nature of a product in the context of a transaction for a tax.

**territory**
A legally distinct region that is used in the country field of an address.

**transaction fiscal classification**
A classification used by a tax authority to categorize a transaction for a tax. There could be more than one by tax. For example, for Brazil, three classifications
are required: a) transaction nature, such as free sample, demonstration, consignment, donation; b) transaction classification, such as the sale of products previously acquired, the sale of products that were manufactured by the company; and c) operation classification, such as ship from - ship to relationship.

**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.

**user rate type**
Rate you enter at journal entry time to convert foreign currency transactions to your functional currency.

**user variable**
A variable that can be created by the Contract Terms Library administrator for use within clause text or in Contract Expert rules.

**value set**
A set of valid values against which values entered by an end user are validated. The set may be tree structured (hierarchical).

**value-added tax (VAT)**
An indirect tax on consumer expenditure that is collected on business transactions and imported goods. Value-added tax (VAT) is charged at each production, distribution, and retail stage in the supply of products. If customers are registered for VAT and use the supplies for taxable business purposes, then they will typically receive credit for the VAT that is paid.

**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.

**worker type**
A classification selected on a person’s work relationship, which can be employee, contingent worker, pending worker, or nonworker.
**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**

Abbreviation for eXtensible markup language.

**XML filter**

A type of condition using XML to constrain the data secured by a data security policy.