## Contents

1 Overview
- Customer Data Management Offering: Overview ............................................................ 1-1
- Getting Started with an Implementation: Overview ......................................................... 1-2
- Manage Application Implementation ............................................................................... 1-4

2 Common Applications Configuration: Define Synchronization of Users and Roles from LDAP
- Define Synchronization of Users and Roles from LDAP ................................................... 2-1

3 Common Applications Configuration: Define Implementation Users
- Define Implementation Users ........................................................................................... 3-1

4 Common Applications Configuration: Define Currencies and Currency Rates
- Manage Currencies ........................................................................................................... 4-1
- FAQs for Manage Currencies .......................................................................................... 4-2
- Manage Conversion Rate Types ....................................................................................... 4-3
- FAQs for Manage Conversion Rate Types ........................................................................ 4-6

5 Common Applications Configuration: Define Enterprise Structures for Customer Data Management
- 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 ....................... 5-12
- FAQs for Define Initial Configuration .............................................................................. 5-37
- Define Reference Data Sharing .......................................................................................... 5-38
- Define Enterprise: Manage Locations .............................................................................. 5-42
- FAQs for Manage Locations ............................................................................................ 5-43
- Define Geographies .......................................................................................................... 5-44
- FAQs for Define Geographies ........................................................................................... 5-64
- Define Legal Entities for Customer Data Management: Manage Legal Entity ................ 5-66
- Define Legal Entities for Customer Data Management: Manage Legal Entity HCM Information .............................................................................................................................. 5-71
- FAQs for Manage Legal Entity HCM Information ............................................................ 5-76
Define Business Units for Customer Data Management: Manage Service Provider Relationships .......................................................... 5-76
Define Business Units for Customer Data Management: Specify Customer Contract Management Business Function Properties ......................................... 5-79
Define Business Units for Customer Data Management: Specify Supplier Contract Management Business Function Properties .................................. 5-83
Define Business Units for Customer Data Management: Manage Business Units Assign Business Unit Business Function .............................................................................................................................. 5-86
Define Workforce Structures for CRM: Manage Locations .............................................................................................................. 5-89
Define Workforce Structures for CRM: FAQs for Manage Locations ................................................................................................. 5-90
Define Workforce Structures for CRM: Manage Divisions .................................................................................................................. 5-91
Define Workforce Structures for CRM: Manage Departments ......................................................................................................... 5-94
Define Workforce Structures for CRM: FAQs for Manage Job Families ............................................................................................. 5-97
Define Workforce Structures for CRM: Manage Jobs ......................................................................................................................... 5-97

6 Common Applications Configuration: Define Security for CRM

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 ................................................. 6-8
Security Tasks: Overview ........................................................................ 6-11
Define Data Security .................................................................................. 6-16
Manage Role Templates .......................................................................... 6-29
Manage Data Role and Security Profiles .................................................... 6-31
Define Users .............................................................................................. 6-38
Import Worker Users ............................................................................... 6-48
Manage Users ........................................................................................... 6-55
FAQs for Manage Users ........................................................................... 6-60

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 Customer Data Management

Approval Management: Highlights ............................................................. 8-1

9 Common Applications Configuration: Define Help Configuration

Define Help Configuration: Overview ........................................................ 9-1
10 Common Applications Configuration: Define Application Toolkit Configuration

Define Application Toolkit Configuration: Overview ........................................ 10-1
Map Reports to Work Areas ........................................................................ 10-1
FAQs for Map Reports to Work Areas ....................................................... 10-2
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 ......................................................................... 11-7
FAQs for Define ISO Reference Data ............................................................ 11-9
Manage Data Security Policies ................................................................... 11-11
Manage Public Unique Identifier Profile Options ...................................... 11-20
Set Activity Stream Options ....................................................................... 11-22
Manage Menu Customizations ................................................................... 11-23
Manage Audit Policies ................................................................................ 11-29
FAQs for Manage Audit Policies after Manage Audit Policies ..................... 11-40
Manage Oracle Social Network Objects ...................................................... 11-45
FAQs for Manage Oracle Social Network Objects ....................................... 11-49
Manage Applications Core Common Reference Objects ............................ 11-51
FAQs for Manage Applications Core Administrator Profile Values .......... 11-53

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 Common Applications Configuration: Define Social Networking

Define Social Networking ............................................................................. 13-1
Administer Oracle Social Network .............................................................. 13-14
FAQs for Define Social Networking ............................................................. 13-21

14 Common Applications Configuration: Define Applications Core Configuration

Define Applications Core Configuration: Overview .................................... 14-1
Define Lookups ........................................................................................... 14-1
15 Common Customer Data Management Configuration: Define Source Systems

Source Systems: Explained ................................................................. 15-1
Source System Entities: Explained ..................................................... 15-2
FAQs for Define Source Systems ....................................................... 15-2

16 Common Customer Data Management Configuration: Define Party Usages

Party Usage: Explained .................................................................... 16-1
Party Usage Filter Rules: Explained .................................................. 16-2
FAQs for Define Party Usages ............................................................ 16-2

17 Common Customer Data Management Configuration: Define Names

Additional Name Types: Explained ..................................................... 17-1

18 Common Customer Data Management Configuration: Define Identifiers

Identifier Types: Explained ............................................................... 18-1

19 Common Customer Data Management Configuration: Define Relationships

Relationships Types: Explained ......................................................... 19-1
Relationship Groups: Explained ....................................................... 19-2
Manage Relationship Types ............................................................. 19-2
Manage Relationship Groups ......................................................... 19-5

20 Common Customer Data Management Configuration: Define Classifications

Classifications: Explained ............................................................... 20-1
FAQs for Define Classifications ....................................................... 20-2

21 Common Customer Data Management Configuration: Define Data Formats

Data Formats and Setup Activities: How They Fit Together .................. 21-1
Setup Formats: Explained ............................................................... 21-2
Creating an Address Style Format: Worked Example ..................... 21-2
Creating a Name Style Format: Worked Example ......................... 21-4
22 Common Customer Data Management Configuration: Define Contact Points

Telephone Numbering Plan: Explained ................................................................. 22-1
Updating Telephone Numbering Plans: Explained ........................................... 22-1
How Setup Telephone Numbering Plans Fits in with the Other Setup Tasks .......... 22-2
FAQs for Define Contact Points ........................................................................... 22-3

23 Common Customer Data Management Configuration: Define Data Import

Data Import Objects: Explained ........................................................................... 23-1
Using Data Quality Services During Data Import: Explained ......................... 23-1
Defining the Import Process for Customers and Consumers: Points to Consider ... 23-2
Defining the Import Process for Customers and Consumers: Worked Example ... 23-4
Defining the Import Process for Resources and Partners: Points to Consider ...... 23-7
Defining the Import Process for Resources and Partners: Worked Example ... 23-8
Performing What-If Analysis on Data Import Batches: Worked Example ........... 23-9
FAQs for Define Data Import ............................................................................. 23-11

24 Common Customer Data Management Configuration: Define File-Based Data Import

File-Based Data Import: Highlights ..................................................................... 24-1
FAQs for Define File-Based Data Import ............................................................. 24-2

25 Common Customer Data Management Configuration: Define Resource Information

Resource Directory: Explained ........................................................................... 25-1
Setting up Resources: Explained ........................................................................ 25-1
Identifying Resources: Explained ....................................................................... 25-2
Resources: How They Work within a Team ......................................................... 25-2
FAQs for Define Resource Information ............................................................... 25-3

26 Common Customer Data Management Configuration: Define Resource Role Information

Define Resource Roles: Explained ....................................................................... 26-1
Provisioning Security Roles: Explained ............................................................... 26-2
Job-to-Role Mapping: Explained ......................................................................... 26-2
Resource Role Assignment: Explained ................................................................. 26-2
Resource Team Membership and Role Assignment Components: How They Work Together ......................................................................................... 26-3
FAQs for Define Resource Role Information ......................................................... 26-3

27 Common Customer Data Management Configuration: Define Resource Organization Information
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Organizations and Organization Usage: Explained</td>
<td>27-1</td>
</tr>
<tr>
<td>Primary Resource Organization by Usage Explained</td>
<td>27-1</td>
</tr>
<tr>
<td>Resource Organization Hierarchy Versioning: Explained</td>
<td>27-1</td>
</tr>
<tr>
<td>Organization Memberships and Role Assignments: Explained</td>
<td>27-2</td>
</tr>
<tr>
<td>Create an Organization with Sales Usage: Worked Example</td>
<td>27-2</td>
</tr>
<tr>
<td>Adding a Sales Organization to a Sales Organization Hierarchy: Worked Example</td>
<td>27-3</td>
</tr>
<tr>
<td>Assigning a Resource to an Organization: Example</td>
<td>27-4</td>
</tr>
<tr>
<td>FAQs for Define Resource Organization Information</td>
<td>27-6</td>
</tr>
<tr>
<td>28 Common Customer Data Management Configuration: Define Resource Team Information</td>
<td>28-1</td>
</tr>
<tr>
<td>Resource Setup and Resource Role Setup: How They Fit Together</td>
<td>28-1</td>
</tr>
<tr>
<td>FAQs for Define Resource Team Information</td>
<td>28-1</td>
</tr>
<tr>
<td>29 Common Customer Data Management Configuration: Manage Calendar Profile Option</td>
<td>29-1</td>
</tr>
<tr>
<td>Managing Accounting Calendars</td>
<td>29-1</td>
</tr>
<tr>
<td>30 Common Customer Data Management Configuration: Manage Bulk Data Export</td>
<td>30-1</td>
</tr>
<tr>
<td>Bulk Export: Overview</td>
<td>30-1</td>
</tr>
<tr>
<td>Bulk Export Process Definition: Explained</td>
<td>30-2</td>
</tr>
<tr>
<td>Bulk Export Process Components: How They Work Together</td>
<td>30-4</td>
</tr>
<tr>
<td>FAQs for Manage Bulk Data Export</td>
<td>30-6</td>
</tr>
<tr>
<td>31 Enriching Sales Cloud Data Using Oracle Social Data and Insight Cloud Service</td>
<td>31-1</td>
</tr>
<tr>
<td>Enriching Account and Account Contact Data: Explained</td>
<td>31-1</td>
</tr>
<tr>
<td>Importing New Records into Oracle Sales Cloud from Oracle Social Data and Insight Cloud Service: Explained</td>
<td>31-2</td>
</tr>
<tr>
<td>32 Define Data Quality</td>
<td>32-1</td>
</tr>
<tr>
<td>Define Data Quality: Explained</td>
<td>32-1</td>
</tr>
<tr>
<td>Manage Server Configurations</td>
<td>32-3</td>
</tr>
<tr>
<td>Manage Matching Configurations</td>
<td>32-5</td>
</tr>
<tr>
<td>Manage Cleansing Configurations</td>
<td>32-15</td>
</tr>
<tr>
<td>Manage Data Synchronization</td>
<td>32-20</td>
</tr>
<tr>
<td>FAQs for Define Data Quality</td>
<td>32-26</td>
</tr>
<tr>
<td>33 Define Customer Hub Configuration</td>
<td>33-1</td>
</tr>
<tr>
<td>Run Request Dispatch Job</td>
<td>33-1</td>
</tr>
<tr>
<td>Manage Customer Hub Profile Options</td>
<td>33-1</td>
</tr>
<tr>
<td>Define Survivorship</td>
<td>33-3</td>
</tr>
<tr>
<td>Manage Party Center Trees</td>
<td>33-8</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Manage Customer Data Hub Notes</td>
<td>33-10</td>
</tr>
<tr>
<td>Manage Agreement Rules</td>
<td>33-11</td>
</tr>
<tr>
<td>Enter Merge Request</td>
<td>33-15</td>
</tr>
</tbody>
</table>

### 34 Define Extensions for Customer Data Management

- Define Custom Enterprise Scheduler Jobs                                | 34-1 |
- FAQs for Define Extensions for Customer Data Management                | 34-4 |

### 35 Importing and Exporting Setup Data

- Configuration Packages: Explained                                     | 35-1 |
- Exporting and Importing Setup Data: Explained                          | 35-1 |
- Moving Common Reference Objects                                       | 35-2 |
Preface

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

Oracle Fusion Applications Help

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

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.
Customer Data Management Offering: Overview

Using the Customer Data Management business process area, your enterprise can review and define the set up for managing customer information and other business relationships, and define the setup configuration and tasks related to customer data management hub deployment.

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

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

- Data Quality
- Data Quality Matching
- Data Quality Cleansing
- Customer Hub
- Customer Data Management Business Intelligence Analytics

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

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

- Define Common Applications Configuration
- Define Common Customer Data Management Configuration
- Define Data Quality
- Define Customer Hub Configuration
- Define Extensions for Customer Data Management

Define Common Applications Configuration for Customer Data Management
Manage definitions used across offerings, typically applying to multiple products and product families. These definitions include enterprise structures, workforce profiles, security, and approval rules, amongst others.

You can find other information that supports the common implementation tasks in the Oracle Fusion Applications Concepts Guide.
Define Common Customer Data Management Configuration
Define the configuration for common setup such as users, enterprise structures, security, approval management, and common reference objects for customer data management.

Define Data Quality
Define and manage data quality lookup choices, configurations, and schedule and monitor processes used in data matching, cleansing and synchronization.

Define Customer Hub Configuration
Define the setup configuration and tasks related to customer data management hub deployment.

Define Extensions for Customer Data Management
Define extensions such as custom Oracle Enterprise Scheduler jobs.
You can also customize and extend applications using other tools. For more information, see the Oracle Fusion Applications Extensibility Guide.

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.
Manage Application Implementation

Manage Application Implementation: Overview

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

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

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

- **Rapid Start**
  Specific implementations can become templates to facilitate reuse and rapid-start for comparable Oracle Fusion applications across many instances.

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

With Oracle Fusion Functional Setup Manager you can:

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

The following documentation resources are available for learning how to configure Oracle Fusion Applications.
Implementation Projects: Explained

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

Maintaining Setup Data

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

Exporting and Importing

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

Selecting Offerings

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

### Offerings: Explained

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

### Implementation Task Lists

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

### Enabling Offerings

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

### Provisioning Offerings

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

### Options: Explained

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

**Feature Choices: Explained**

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

You can configure feature choices in three different ways:

**Yes or No**

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

**Single Select**

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

**Multi-Select**

If the feature has multiple choices but one or more can be applicable to an implementation then all choices are presented with a checkbox. Select all that apply by checking the appropriate choices.
Common Applications Configuration: Define Synchronization of Users and Roles from LDAP

Define Synchronization of Users and Roles from LDAP

User and Role Synchronization: Explained

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

Note

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

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

Tip

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

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>Decision</td>
<td>In this Example</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</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 administrator 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 Inherits From and click Add.

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

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

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

6. Assign the IT Security Manager role to the Xellerate Users organization.
   a. In the Welcome to Identity Manager Delegated Administration menu (see step 1, above), in the Organizations list of tasks, click Advanced Search - Organizations.
   b. Search for the Xellerate Users organization by entering Xellerate Users in Display Name and clicking Search.
   c. In the Search Results, click the organization’s Display Name. The Xellerate Users page appears.
   d. Click the Administrative Roles link in the row of links above the Xellerate Users.
   e. In Filter By Role Name of the Details window, enter the following string:

      *IT_SECURITY_MANAGER*

   f. Click Find.
   g. Enable Read, Write, Delete, and Assign.
   h. Click Assign.
   i. 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$$w0 Rd</td>
</tr>
</tbody>
</table>

**Note**

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

5. On the Roles tab in the IT_SECURITY_MANAGER user creation task flow, click **Assign**.

6. In the Add Role window, search for the IT Security Manager role and click **Add**.

**Defining an Implementation User for Enterprise Structures Setup**

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

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

   - Application Implementation Manager
   - Application Implementation Consultant

**Note**

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

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>
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.
Enterprise Structures: Overview

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

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

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

The figure above shows a typical group of legal entities, operating various business and functional organizations. Your ability to buy and sell, own, and employ comes from your charter in the legal system. A corporation is a distinct legal entity from its owners and managers. The corporation is owned by its shareholders, who may be individuals or other corporations. There are many other kinds of legal entities, such as sole proprietorships, partnerships, and government agencies.

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

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

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

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

Management Structure

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

Functional Structure

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

**Enterprise Structures Business Process Model: Explained**

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

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

<table>
<thead>
<tr>
<th>BPM Activities</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define Facilities</td>
<td>Define inventory, item, and cost organizations. Inventory organizations represent facilities that manufacture or store items. The item master organization holds a single definition of items that can be shared across many inventory organizations. Cost organizations group inventory organizations within a legal entity to establish the cost accounting policies.</td>
</tr>
<tr>
<td>Define Reference Data Sharing</td>
<td>Define how reference data in the applications is partitioned and shared.</td>
</tr>
</tbody>
</table>

**Note**

There are product specific implementation activities that are not listed here and depend on the applications you are implementing. For example, you can implement Define Enterprise Structures for Human Capital Management, Project Management, and Sales Management.

## Global Enterprise Configuration: Points to Consider

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

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

### Enterprise Configuration

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

### Business Unit Management

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

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

Compliance Requirements

How do you comply with your corporate external reporting requirements and local statutory reporting requirements? Do you tend to prefer a corporate first or an autonomous local approach? Where are you on a spectrum of centralization, very centralized or decentralized?

Modeling Your Enterprise Management Structure in Oracle Fusion: Example

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

Scenario

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

InFusion Corporation

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

Analysis

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

- Your company is required to report using US 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>Oracle Fusion General Ledger Configuration</td>
<td>Maps to Essbase As:</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------------------------</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>
<tr>
<td>.</td>
<td>period</td>
</tr>
<tr>
<td>+</td>
<td>plus sign</td>
</tr>
<tr>
<td>'</td>
<td>single quotation mark</td>
</tr>
<tr>
<td>_</td>
<td>underscore</td>
</tr>
<tr>
<td></td>
<td></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>
</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.
**Division: Explained**

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

**Divisions**

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

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

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

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

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

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

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

**Legal Entities: Explained**

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

![Diagram of InFusion Corporation’s enterprise structure](image)

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>Business Unit Level</td>
<td>Suggested Business Units</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>----------------------------------------------------------------</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>
<tr>
<td>Division and Business Function</td>
<td>InFusion Lighting, Sales</td>
</tr>
<tr>
<td></td>
<td>InFusion Lighting, Marketing</td>
</tr>
<tr>
<td></td>
<td>Infusion Security, Sales</td>
</tr>
<tr>
<td></td>
<td>Infusion Security, Marketing</td>
</tr>
<tr>
<td>Business Function</td>
<td>Sales</td>
</tr>
<tr>
<td></td>
<td>Marketing</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Legal Entity: Japan</td>
</tr>
<tr>
<td></td>
<td>Legal Entity: US</td>
</tr>
<tr>
<td></td>
<td>Legal Entity: UK</td>
</tr>
<tr>
<td></td>
<td>Legal Entity India</td>
</tr>
<tr>
<td>Legal Entity and Business Function</td>
<td>Legal Entity: Japan, Sales</td>
</tr>
<tr>
<td></td>
<td>Legal Entity: Japan, Marketing</td>
</tr>
<tr>
<td></td>
<td>Legal Entity: US, Sales</td>
</tr>
<tr>
<td></td>
<td>Legal Entity: UK, Sales</td>
</tr>
<tr>
<td></td>
<td>Legal Entity India, Marketing</td>
</tr>
<tr>
<td></td>
<td>Legal Entity India, Sales</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 instance, Payables Payment Terms use this method. It means that each payment term can be assigned to one or more than one set. For example, you assign the payment term Net 30 to several sets, but the payment term Net 15 is assigned to only your corporate business unit specific set. At transaction entry, the list of values for payment terms consists of only one set of data; the set that is assigned to the transaction’s business unit.

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

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

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

**Common Set Versus Specific Sets**

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

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

**Business Unit Set Assignment**

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

**Example: Assigning Sets to Business Units**

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

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

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

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

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

InFusion assigns business units to sets as follows:

<table>
<thead>
<tr>
<th>Business Unit</th>
<th>Default Set Assignment</th>
<th>Set Assignment Overrides</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales_Japan</td>
<td>Sales_Japan_Set for grades, departments, and locations</td>
<td>Jobs set for jobs</td>
</tr>
<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>
</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 and Warehousing</td>
<td>Positions</td>
</tr>
<tr>
<td>Agriculture, Forestry, Fishing, and Hunting</td>
<td>Jobs</td>
</tr>
</tbody>
</table>
### Primary Industry  |  Workforce Setup
---|---
Construction | Jobs
Wholesale Trade | Jobs
Information | Jobs
Finance and Insurance | Jobs
Professional, Scientific, and Technical Services | Jobs
Management of Companies and Enterprises | Jobs
Administrative and Support and Waste Management and Remediation Services | Jobs
Arts, Entertainment, and Recreation | Jobs
Accommodation and Food Services | Jobs
Other Services (Except Public Administration) | Jobs

## 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>
</tbody>
</table>
### Positions: Examples

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

**Retail Industry**

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

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

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

This figure illustrates the retail position setup.
Health Care Industry

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

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

This figure illustrates the hospital position setup.

Jobs: Example

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

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

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

This figure illustrates the software industry job setup.

Job and Position Structures: Explained

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

Enterprise-Level Job Attributes

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

This figure illustrates how job type and job level provide further details for the HR Application Specialist job.
Enterprise-Level Position Attributes

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

This figure illustrates how title and position number provide further details for the manager position.

Business Unit-Level Attributes for Positions

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

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

**FAQs for Define Initial Configuration**

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

The Enterprise Structures Configurator is an interview-based tool that guides you through setting up divisions, legal entities, business units, and reference data sets. The tool also enables you to assign reference data sets to business units and locations. You can set up multiple configurations to perform what-if scenarios, and then print each configuration to compare the resulting enterprise structure. If you do not use the Enterprise Structures Configurator, then you must set up your enterprise structure using the individual tasks that correspond to each enterprise component. In addition, you will not be able to set up multiple configurations and compare different scenarios. It is recommended that you use the Enterprise Structures Configurator.

**What's an ultimate holding company?**

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

**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
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 Sharing Diagram](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.

### 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.

**Geocoding: Explained**

Geocoding is the process of finding latitude and longitude coordinates from geographic data such as street addresses or zip codes. Once these coordinates are available, you can use the spatial services feature to identify points of interest, such as customer and contact addresses, in the vicinity. Oracle Fusion Applications integrate the Geocoding feature with eLocation, a geocoder service provided by Oracle.

If the Geocoding feature is enabled, the feature can be scheduled to run at regular time intervals so that newly created or updated locations are picked up and geocoded whenever you create or update an address using the user interface, web services, bulk import, or file-based import. By default, the Geocoding option is turned off in Oracle Fusion Applications. You can enable geocoding by using the **Setup and Maintenance, Manage Geographies** page. The feature can be scheduled to run at regular time interval so that newly created or updated locations are picked up and geocoded.

**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_T</td>
<td>HZ_Geographies</td>
<td>Geography</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_Geography_IDentiFiers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_Geography_Types_B</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HZ_Hierarchy_Nodes</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://fusionappssoer.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>ImpGeoStructur eLevel</td>
<td>Contains information that specifies a country’s geography structure. Sample attributes: GeographyType, GeographyTypeName, LevelNumber, and ParentGeographyType. Reference attribute: CountryCode</td>
<td>HZ_IMP_GEO_STRUCTURE_ LEVELS_Reference</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 object. 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
Oracle Sales Cloud Implementing Customer 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).

5-56 Oracle Sales Cloud Implementing Customer Data Management
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.</td>
<td>HZ_IMP_GEOGRAPHIES_T_Reference</td>
</tr>
<tr>
<td></td>
<td>Sample attributes: CountryCode, GeoDataType, GeographyType,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PrimaryGeographyCode, PrimaryGeographyCodeType, and PrimaryGeographyName.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Reference attribute: CountryCode</td>
<td></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>
<tr>
<td>Which fields are you importing into Oracle Sales Cloud?</td>
<td>All, except for the RecordTypeCode field</td>
</tr>
<tr>
<td>When do you want to process the import?</td>
<td>Immediately</td>
</tr>
</tbody>
</table>

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>
</tbody>
</table>
### Column Headers, Example Values, Ignore, Object, Attribute

<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>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>What is the structure of the geography types?</td>
<td>Create geography types with the following ranking structure:</td>
</tr>
<tr>
<td></td>
<td>County</td>
</tr>
<tr>
<td></td>
<td>Post Town</td>
</tr>
<tr>
<td>What is the geography hierarchy?</td>
<td>Create the following hierarchy:</td>
</tr>
<tr>
<td></td>
<td>Country of United Kingdom</td>
</tr>
<tr>
<td></td>
<td>County of Berkshire</td>
</tr>
<tr>
<td></td>
<td>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.

How can I verify whether the third-party geography structure and hierarchy I imported are available in the application?

You can track the progress of the geography structure and hierarchy import process by navigating to the Scheduled Processes page and viewing the status of the third-party geography data import process. Once the process completes, the status changes to Succeeded. You can then navigate to the Setup and Maintenance, Manage Geographies page, where you first loaded the third-party geography data, to verify the newly imported geography structure and hierarchy.

Note
Before you load geography data for a country from a third party provider, ensure that no geography structure or hierarchy is saved for that country. If geography structure or hierarchy data is already available, the load process fails.

How can I enable geocoding?

You can enable geocoding by turning the Geocoding option on in the Setup and Maintenance, Manage Geographies page. Search for the country for which you want to enable geocoding and click the Geocoding Defined icon. If geocoding is enabled for a country, the Geocoding Defined icon displays a check mark.

Once you have enabled geocoding for a country, you need to run the Run Geocode Generation ESS job. To do so, navigate to Setup and Maintenance, Run Geocode Generation. Before you run the job, however, make sure that you have specified the country code of the country for which you wish to generate the geocodes.

What are Spatial Services?

Spatial services allow users to find points of interest such as customers, contacts, and so on using the latitude and longitude coordinates of an address.

The Oracle Fusion Mobile Sales application provides these services for sales representatives to discover accounts and contacts around their location.

Where can I update and view the geography name reference information for parties?

To view the geography name reference information associated with parties, you need to ensure that you have created an address for the party. You can create or update party addresses in the Party Center and the Customer Center pages.
Once you have updated an address, you either need to either wait for the Run Geocode Generation ESS job to run automatically as scheduled or start geocode generation manually from the Setup and Maintenance > Run Geocode Generation page.

Once the Run Geocode Generation job has updated the geography information for the country associated with the party you updated, you can search for and view the geography name reference information associated with the party in the Party Center and the Customer Center pages. Navigate to the person, organization, or group whose address you want to verify and click on Actions, View Geography Information. This displays a popup that displays the updated geography information, complete with address values enhanced using geography name reference.

**Note**

Geography name reference enriched geography information display is available only when geography validation is enabled for the country to which the address belongs.

---

**Define Legal Entities for Customer Data Management: 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 for Customer Data Management: 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 Business Units for Customer Data Management: 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 for Customer Data Management: 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 **Set up and maintain** 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 for Customer Data Management: 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 Business Units for Customer Data Management: 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.

Associating Business Units with Resource Organizations: Explained

A business unit performs one or more enterprise business functions and can be rolled up in a management hierarchy. Associating resource organizations with business units helps controlling access to contracts based on business units.

Use the HZ_DEFAULT_BU_CRM profile option to enable business units in Oracle Fusion Applications, and the HZ_ENABLE_MULTIPLE_BU_CRM profile option to specify whether you want to enable multiple business units for Oracle Fusion Enterprise Contracts. The first business unit that you associate with a resource organization will automatically be listed as the primary unit.

Define Business Units for Customer Data Management: 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 Workforce Structures for CRM: 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 for CRM: 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 for CRM: 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>Decision to Consider</td>
<td>In This Example</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------</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. 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. 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.</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>

**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 for CRM: 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 for CRM: 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 for CRM: Manage Jobs

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.
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.
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.

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.

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.

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

• You manage duties, security policies, and data role templates in the Authorization Policy Manager. Refer to the Oracle Fusion Middleware Authorization Policy Manager Administrator's Guide.

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 (OIM). 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
  
  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</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>role</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add a job role to a role hierarchy</td>
<td>Manage Job Roles</td>
<td>Oracle Identity Management (OIM)</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Add a duty role to a role hierarchy</td>
<td>Manage Duties</td>
<td>APM</td>
<td>LDAP stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions.</td>
</tr>
<tr>
<td>Create a hierarchy of enterprise (abstract, job, data roles)</td>
<td>Manage Job Roles</td>
<td>OIM</td>
<td></td>
</tr>
<tr>
<td>Create a hierarchy of (application) duty roles</td>
<td>Manage Duties</td>
<td>APM</td>
<td></td>
</tr>
<tr>
<td>Create a new job role</td>
<td>Manage Job Roles</td>
<td>OIM</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Change duty roles inherited by a job or abstract role</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>Example Task</td>
<td>Oracle BPM Task</td>
<td>Supporting Tools</td>
<td>Details</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create a new duty role</td>
<td>Manage Duties</td>
<td>APM</td>
<td>All functions and actions in Oracle Fusion Applications that need to be secured are covered by the reference implementation. In some cases, especially with function customizations, a new duty role may be needed.</td>
</tr>
<tr>
<td>View Segregation of Duties (SOD) policies respected by a duty role</td>
<td>Manage Application Access Controls</td>
<td>Application Access Controls Governor (AACG) in Oracle Enterprise Governance, Risk and Compliance (GRC)</td>
<td>The Security Reference Manuals (SRM) document the segregation of duties (SOD) policies respected within 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>
</tbody>
</table>
| View the data security policies carried by a job, abstract, and data roles | Manage Data Security Policies | APM              | Oracle Fusion Data Security stores data security policies in the policy store.  
Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)                                                                                     |
<p>| Create and update HCM security profiles                                   | 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.                                                                                                     |</p>
<table>
<thead>
<tr>
<th>Example Task</th>
<th>Oracle BPM Task</th>
<th>Supporting Tools</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create (generate) a data role</td>
<td>Manage Role Templates</td>
<td>APM</td>
<td>Data roles are generated automatically based on data role templates and enterprise setup. Changes to data role templates generate new or changed data roles. Create data roles in HCM using the Manage Data Roles and Security Profiles task.</td>
</tr>
<tr>
<td>Create a new data security policy (not through generated data roles based on data role templates or HCM security profiles)</td>
<td>Manage Data Security Policies</td>
<td>APM</td>
<td>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>View data role templates defined by a product</td>
<td>Manage Role Templates</td>
<td>APM</td>
<td></td>
</tr>
<tr>
<td>Create or edit an existing data role template</td>
<td>Manage Role Templates</td>
<td>APM</td>
<td></td>
</tr>
<tr>
<td>Secure common objects such as attachment categories or profile options</td>
<td>Manage Data Security Policies</td>
<td>APM</td>
<td>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>View, create, update Data Access Sets used to secure Ledgers and Ledger Sets</td>
<td>Manage Data Access Sets</td>
<td>Oracle Fusion General Ledger</td>
<td></td>
</tr>
<tr>
<td>View, create, update accounting flexfield segment security rules</td>
<td>Manage Security Segments</td>
<td>Oracle Fusion General Ledger</td>
<td></td>
</tr>
<tr>
<td>View or update the set of job roles that can be provisioned to supplier users</td>
<td>Manage Supplier User Role</td>
<td>Supplier Portal Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Determine the supplier job roles that the supplier self service administrator can provision to supplier users</td>
<td>Manage Supplier User Role Usages</td>
<td>Supplier Portal Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Example Task</td>
<td>Oracle BPM Task</td>
<td>Supporting Tools</td>
<td>Details</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>--------------------------------------</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</td>
<td>Supplier Portal Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family</td>
</tr>
<tr>
<td>Create a new implementation user</td>
<td>Create Implementation Users</td>
<td>OIM</td>
<td></td>
</tr>
<tr>
<td>Import legacy users</td>
<td>Import Worker Users</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>HCM creates a new user and identity when a new worker is created. The Hire Employee and Add Contingent Worker tasks also result in new user creation requests. Creating a new user automatically triggers role provisioning requests based on role provisioning rules.</td>
</tr>
<tr>
<td>Example Task</td>
<td>Oracle BPM Task</td>
<td>Supporting Tools</td>
<td>Details</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>-----------------</td>
<td>---------</td>
</tr>
<tr>
<td>Provision roles to a user</td>
<td>Provision Roles to Implementation Users Manage Users</td>
<td>OIM Oracle Fusion HCM Oracle Sales Cloud Oracle Fusion Suppliers</td>
<td>Implementation users are provisioned through OIM since HCM is not setup at the start of the implementation. The Provision Roles to Implementation Users is not needed once implementation is complete. Once HCM is setup, HCM is used to provision roles to non-implementation users by performing the Manage Users task. Human Resources (HR) transaction flows such as Hire and Promote also provision roles. Once supplier users are setup, Supplier Model can be used by internal users to maintain supplier user accounts or supplier users can maintain their accounts in Supplier Portal.</td>
</tr>
<tr>
<td>View the job, abstract, and data roles provisioned to a user</td>
<td>Manage Users Manage User Principal Provision Roles to Implementation Users</td>
<td>Human Capital Management (HCM) OIM</td>
<td>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.</td>
</tr>
<tr>
<td>Revoke role from user.</td>
<td>Manage Users</td>
<td>HCM</td>
<td>You can revoke roles from various Human Resources task flows, the HCM Manage Users task and OIM. User termination includes role revocation.</td>
</tr>
<tr>
<td>Approve role provisioning or user account request.</td>
<td>Approve User and Role Provisioning Requests</td>
<td>OIM</td>
<td></td>
</tr>
<tr>
<td>Example Task</td>
<td>Oracle BPM Task</td>
<td>Supporting Tools</td>
<td>Details</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------</td>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>View audit logs</td>
<td>Not applicable</td>
<td>Oracle Enterprise Manager</td>
<td>Viewing audit logs is a Oracle Fusion Middleware function and not represented by an Oracle Fusion Applications BPM task.</td>
</tr>
</tbody>
</table>

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.

<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.

**Defining Data Security After Enterprise Setup: Points to Consider**

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise.

The Define Data Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Manage Data Access Sets
- Manage Segment Security
- Manage Role Templates
- Manage Data Security Policies
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.

**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.

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 inher iting job and abstract roles with implicit data security</td>
<td>Projects</td>
</tr>
</tbody>
</table>

If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

When a runtime session is created, Oracle Platform Security Services (OPSS) propagates only the necessary user to role mapping based on Oracle Fusion Data Security grants. A grant can specify entitlement to the following.

- Specific rows of data (data object) identified by primary key
- Groups of data (instance set) based on a predicate that names a particular parameter
- Data objects or instance sets based on runtime user session variables

Data is either identified by the primary key value of the row in the table where the data is stored. Or data is identified by a rule (SQL predicate) applied to the WHERE clause of a query against the table where the data is stored.

Grants

Oracle Fusion Data Security can be used to restrict the following.

- Rows that are returned by a given query based on the intended business operation
• Actions that are available for a given row
Grants control which data a user can access.

Note
Attribute level security using grants requires a data security policy to secure the attribute and the entitlement check enforces that policy.

A grant logically joins a user or role and an entitlement with a static or parameterized object instance set. For example, REGION='WEST' is a static object instance set and REGION=&GRANT_ALIAS.PARAMETER1 is a parameterized object instance set. In the context of a specific object instance, grants specify the allowable actions on the set of accessible object instances. In the database, grants are stored in FND_GRANTS and object instance sets are stored in FND_OBJECT_INSTANCE_SETS. Object access can be tested using the privilege check application programming interface (API).

Securing a Business Object

A business object is a logical entity that is typically implemented as a table or view, and corresponds to a physical database resource. The data security policies of the security reference implementation secure predefined database resources. Use the Manage Data Security Policies task to define and register other database resources.

Data security policies identify sets of data on the registered business object and the actions that may be performed on the business object by a role. The grant can be made by data instance, instance set or at a global level.

Note
Use parameterized object instance sets whenever feasible to reduce the number of predicates the database parses and the number of administrative intervention required as static object instances sets become obsolete. In HCM, security profiles generate the instance sets.

Database Resources and Data Security Policies: How They Work Together

A data security policy applies a condition and allowable actions to a database resource for a role. When that role is provisioned to a user, the user has access to data defined by the policy. In the case of the predefined security reference implementation, this role is always a duty role. Data roles generated to inherit the job role based on data role templates limit access to database resources in a particular dimension, such as the US business unit.

The database resource defines and instance of a data object. The data object is a table, view, or flexfield.

The following figure shows the database resource definition as the means by which a data security policy secures a data object. The database resource names the data object. The data security policy grants to a role access to that database resource based on the policy’s action and condition.
Database Resources

A database resource specifies access to a table, view, or flexfield that is secured by a data security policy.

- Name providing a means of identifying the database resource
- Data object to which the database resource points

Data Security Policies

Data security policies consist of actions and conditions for accessing all, some, or a single row of a database resource.

- Condition identifying the instance set of values in the data object
- Action specifying the type of access allowed on the available values

Note

If the data security policy needs to be less restrictive than any available database resource for a data object, define a new data security policy.

Actions

Actions correspond to privileges that entitle kinds of access to objects, such as view, edit, or delete. The actions allowed by a data security policy include all or a subset of the actions that exist for the database resource.

Conditions

A condition is either a SQL predicate or an XML filter. A condition expresses the values in the data object by a search operator or a relationship in a tree.
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 define data security policies. They enable users to perform tasks, using identified menus, menu items, and pages in application user interfaces, on a specified set of data. Because data roles are specific to the enterprise, no predefined HCM data roles exist.

To create an HCM data role, you need the IT Security Manager job role. You perform the task Manage Data Role and Security Profiles (Navigator - Tools - Setup and Maintenance - Manage Data Role and Security Profiles).

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 object types 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, then users with that job role need to access managed person, public person, organization, position, LDG, and document type data. Therefore, you must include security profiles for those object types in the HCM data role. The security profile types that appear in the HCM data role vary with 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, then you can’t create an HCM data role.

Note

If you create custom job roles in OIM, then you must add them to a locally defined role category that ends with Job Roles. Otherwise, they don’t appear in the list of job roles when you create an HCM data role. Don’t 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 for each role remain separate. For example, a user who can promote workers in the Purchasing Department with one HCM data role and view details of workers
in the Sales Department with another can’t promote 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 data security policies for the data role.

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 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, such as person type, that identify one or more person records. When you include a security profile in an HCM data role and provision the role to a user, that user can access the data identified in the security profile. The type of access available to the user (for example whether the user can edit it) depends on the job role identified in the HCM data role.

To create a security profile, you need the IT Security Manager job 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

**Security Criteria in HCM Security Profiles**

In a 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. All criteria in a security profile apply. For example, if you identify organizations by both organization hierarchy and organization classification, then only organizations that satisfy both criteria belong to the data instance set.

**Predefined HCM Security Profiles**

The following HCM security profiles are predefined:

<table>
<thead>
<tr>
<th>Security Profile Name</th>
<th>HCM Security Profile Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View All People</td>
<td>Person</td>
<td>All person records in the enterprise</td>
</tr>
<tr>
<td>View Own Record</td>
<td>Person</td>
<td>The signed-in user’s own person record and the person record s of that user's contacts</td>
</tr>
<tr>
<td>View Manager Hierarchy</td>
<td>Person</td>
<td>The signed-in user’s line manager hierarchy</td>
</tr>
<tr>
<td>Security Profile Name</td>
<td>HCM Security Profile Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>View All Workers</td>
<td>Person</td>
<td>The person records of all people with a work relationship in the enterprise</td>
</tr>
<tr>
<td>View All Organizations</td>
<td>Organization</td>
<td>All organizations in the enterprise</td>
</tr>
<tr>
<td>View All Positions</td>
<td>Position</td>
<td>All positions in the enterprise</td>
</tr>
<tr>
<td>View All Legislative Data Groups</td>
<td>LDG</td>
<td>All LDGs in the enterprise</td>
</tr>
<tr>
<td>View All Countries</td>
<td>Country</td>
<td>All countries in the FND_TERRITORIES table</td>
</tr>
<tr>
<td>View All Document Types</td>
<td>Document Type</td>
<td>All document types in the enterprise</td>
</tr>
<tr>
<td>View All Payrolls</td>
<td>Payroll</td>
<td>All payrolls in the enterprise</td>
</tr>
<tr>
<td>View All Flows</td>
<td>Payroll Flow</td>
<td>All payroll flows in the enterprise</td>
</tr>
</tbody>
</table>

You can include the predefined security profiles in any HCM data role, but you can’t edit them. The **View all** option is disabled in any security profile that you create. This restriction exists because predefined security profiles meet this requirement.

**Creating Security Profiles**

You can create security profiles either individually or while creating an HCM data role. If you have standard requirements, it’s more efficient to create the security profiles individually and include them in appropriate HCM data roles.

To create security profiles individually, use the relevant security profile task. For example, to create a position security profile, use the task Manage Position Security Profiles (Navigator - Tools - Setup and Maintenance - Manage Position Security Profiles).

**Reusability and Inheritance of Security Profiles**

Regardless of how you create them, all security profiles are reusable. They don’t 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

These abstract roles are predefined in Oracle Fusion Human Capital Management:

- Employee
- Contingent worker
- Line manager

Users with these roles can sign in to Oracle Fusion Applications and open application pages. However, they have no automatic access to data. For example, employees can open the Person Gallery but can’t view portraits. Line managers can open the Manager Resources Dashboard but can’t see data for their organizations. To enable users with abstract roles to access relevant HCM data, you assign security profiles directly to those abstract roles.

Predefined Security Profiles to Assign to Abstract Roles

To enable basic data access for users with abstract roles, 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</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>View All Payrolls</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.

Caution
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 the 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**

To enable basic data access for the predefined employee, contingent worker, and line manager abstract roles, you assign predefined security profiles to the roles during implementation. This example shows how to assign security profiles to abstract roles using the Manage Data Role and Security Profiles task.

### Searching for the Employee Abstract Role

1. In the cloud, sign in as the TechAdmin user. On-premises users must sign in with a role that has the IT Security Manager job role.
2. Open the Setup and Maintenance work area (Navigator - Tools - Setup and Maintenance).
3. On the All Tasks tab of the Overview page of the Setup and Maintenance work area, search for and select the task Manage Data Role and Security Profiles.
4. On the Manage Data Roles and Security Profiles page, enter Employee in the Role field. Click **Search**.
5. In the Search Results region, select the predefined Employee role and click **Edit**.

   The Edit Data Role: Role Details page opens.

### Assigning Security Profiles to the Employee Abstract Role

1. On the Edit Data Role: Role Details page, click **Next**.
2. On the Edit Data Role: Security Criteria page, select the security profiles shown in the following table. You may see a subset of these security profiles, depending on the combination of cloud services or 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>
</tbody>
</table>
3. Click **Review**.

4. On the Edit Data Role: Review page, click **Submit**.

5. On the Manage Data Roles and Security Profiles page, search again for the predefined Employee role.

6. 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 Line Manager in the **Role** field. Click **Search**.

2. In the Search Results region, select the predefined Line Manager role and click **Edit**.

   The Edit Data Role: Role Details page opens.

**Assigning Security Profiles to the Line Manager Abstract Role**

1. On the Edit Data Role: Role Details page, click **Next**.

2. On the Edit Data Role: Security Criteria page, select the security profiles shown in the following table. You may see a subset of these security profiles, depending on the combination of cloud services or product offerings that you are implementing.
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<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</td>
<td>View All Payrolls</td>
</tr>
<tr>
<td>Payroll Flow</td>
<td>View All Flows</td>
</tr>
</tbody>
</table>

3. Click **Review**.

4. On the Edit Data Role: Review page, click **Submit**

5. On the Manage Data Roles and Security Profiles page, search again for the predefined Line Manager role.

6. 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.

**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 **Autoprov**ision 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 autoprovision roles from that user’s account.
Role-Mapping Names

The names of role mappings must be unique in the enterprise. You are recommended to devise a naming scheme that reveals the scope of each role mapping. For example:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoprovioned Roles Sales Department</td>
<td>Mapping includes all roles provisioned automatically to anyone in the sales department</td>
</tr>
<tr>
<td>Benefits Specialist Autoprovioned</td>
<td>Mapping defines the conditions for autoprovioning 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.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>Sales</td>
</tr>
<tr>
<td>Job</td>
<td>Sales manager</td>
</tr>
<tr>
<td>Grade</td>
<td>6</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping, you include the:

- Sales Manager role, and select the 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>WorkersMMDDYYBatchnn.xlsx</td>
</tr>
<tr>
<td>You can define your own naming convention. In this example, the name is selected to make identifying the spreadsheet contents easy.</td>
<td>For example, Workers042713Batch01.xlsx.</td>
</tr>
</tbody>
</table>
Decisions to Consider | In This Example
--- | ---
What is my batch name? | Workers042713B atchnn
You can define your own batch name, which must be unique. In this example, the batch name is the same as the spreadsheet name.

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>Attribute</td>
<td>Value</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</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 **Autoprov**is**ion** option
- Expense Claims Administrator role, and select the **Self-req**uestable 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:

<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>gail. <a href="mailto:williams@vision.com">williams@vision.com</a></td>
</tr>
<tr>
<td>Hire Date</td>
<td>4/12/11</td>
</tr>
</tbody>
</table>

3. In the User Details region, leave the User Name field blank. The user name defaults to the user's e-mail ID.

4. In the Employment Information region, select the person type **Employee** and complete the fields as shown in the table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Legal Employer</td>
<td>Vision Corporation</td>
</tr>
<tr>
<td>Business Unit</td>
<td>Vision Canada</td>
</tr>
<tr>
<td>Department</td>
<td>Human Resources</td>
</tr>
</tbody>
</table>

**Assigning Roles to the User**

1. Click **Autoprov**is**ision 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.
Changing User Names: Explained

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

**Changing User Names When Creating Users**

When you create a user using the Create User interface, the **User Name** field may be blank. If you leave it blank, then the user name is generated in the enterprise default format when you save the user.

Alternatively, you can enter a user name on the Create User page, regardless of whether the field is blank. Any name that you enter replaces the default user name.

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

**Changing Existing User Names**

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

When you change an existing user name:

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

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

**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>Value</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</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 e-mail address. If the user has no primary work-email address, then the user name and password are sent to the primary work e-mail 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.

What happens when I autoprovion roles?

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

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

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

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

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

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

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

If the default user-name format for the enterprise is the person number, then primary e-mail is used instead for party users who have no person number.

Default user names may not appear for new users until the relevant value (for example, the person number) is available. The user-account request is not submitted to OIM until the value exists.

What happens when I link a user account?

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

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

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

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.
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 Manager role entitles the user to enter orders, which could result in exploitation when filling shipping quotas. You can remove the entitlement to enter orders that the Order Manager job role inherits from the Orchestration Order Scheduling Duty role. Or you could segregate the shipping and order entry duties by defining an SOD policy that allows a user to have either job role but not both.

**False Positives**

False positives can be SOD policy violations that are not actually violations, or are violations within your risk tolerance and therefore do not require corrective action.

You can reduce false positives by the following methods.

- Define exclusion conditions that can be applied to individual or groups of policies.
- Define logically complex SOD policies that enforce more exacting specifications.
- Determine whether conflicts should be prevented, monitored, or subjected to approval during provisioning.

**Path Level Detection**

Conflict analysis detects a user’s multiple paths to one or more conflicting access points.

For example, a user may be able to reach a single access point through one or more roles, or by one entitlement leading to another through submenus to a function that represents a risk. The resulting conflict path shows if the conflict is generated by inappropriate role provisioning or configuration of applications. The audit shows the paths from any number of users to any number of access points involved in conflicts, which lets you visualize the root cause and remediate effectively.

AACG assigns one or more users to review all paths involved in a given conflict so that the entire conflict can be addressed in a coherent way.

**Waiving or Accepting Violations**

AACG lets you accept or waive a violation. Your reasons may include that you accept the risk or will define compensating controls.

A waiver may apply to the current user, constraint, or constraint within a dimension such as the business unit.
Resolving Conflicts

The risk tolerance of the enterprise determines whether a segregation of duties conflict must be removed from the security reference implementation.

The following approaches resolve conflicts.

• Change the segregation of duties policy.

• Ensure a job role does not contain incompatible duties.

• Define data security policies that restrict authorized access by incompatible duties.

Changing a segregation of duties policy may not be possible in most cases. For example, a policy that segregates creation of payables invoice from making payables payments should be preserved, even if the Accounts Payables Manager job role includes a duty role for each activity. To prevent an accounts payables manager from being authorized to perform both duties, or from being authorized to make payables payments to self and direct reports, the Accounts Payables Manager job role must be changed. The security implementation can be changed to include two job roles that segregate the incompatible duties. Added data security policy grants can restrict the access to at risk data.

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

Role Provisioning and Segregation of Duties: How They Work Together

Segregation of duties (SOD) checks occur when roles are assigned to users. The checks are based on Oracle Application Access Controls Governor (AACG) policies in Oracle Enterprise Governance, Risk and Compliance (GRC). The Oracle Identity Management (OIM) integration includes predefined routing rules for remediation in the Manage IT Security business process.

External users such as suppliers or partners need to be provisioned with roles to facilitate access to parent company interfaces and data. The process by which such provisioning requests are approved in Oracle Fusion Applications helps explain the request flows and possible outcomes.

Note

In Oracle Identity Management (OIM), external users means users who are not specific to applications, such as enterprise roles or the absence of entitlement to access an application.
The figure shows the role provisioning request flow. OIM uses AACG to check segregation of duties violations.

Tables

A supplier or partner requests admission to a program using an implementation of the Supplier Portal Submission. The submission is captured in one or both of the following tables in advance of approving or rejecting the supplier or partner.

- Oracle Fusion Trading Community Model
- Interface Staging

Oracle Fusion Applications collects the employee names for the supplier or partner company at the time the company submits its request to join the program so that all employees accessing Oracle Fusion Applications on behalf of the supplier or partner are provisioned.

AACG in the Oracle Enterprise Governance, Risk and Compliance (GRC) suite is certified to synchronize with the policy and identity stores for all pillars or partitions of Oracle Fusion Applications and integrated with the Oracle Fusion Applications security approach to roll up entitlements (by means of duty roles) to the roles that are provisioned to internal users. SOD policies can be defined and enforced at any level of authorization. For external users, SOD policies use attribute information stored in the Trading Community Model tables.

OIM and the SPML Client

Enterprise business logic may qualify the requester and initiate a role provisioning request by invoking the Services Provisioning Markup Language.
(SPML) client module, as may occur during onboarding of internal users with Human Capital Management (HCM), in which case the SPML client submits an asynchronous SPML call to OIM. Or OIM handles the role request by presenting roles for selection based on associated policies.

OIM recognizes the role provisioning request and initiates a call to AACG.

OIM apprises the SPML client of the current state of the role provisioning request as SOD_CHECK_IN_PROGRESS.

OIM stores the SOD check result as part of OIM audit data.

OIM apprises SPML client of the current state of the SPML request. The provisioning is either still in progress with segregation of duties being checked, or conflicts were found. If conflicts exist, AACG rejects the request and notifies the application.

<table>
<thead>
<tr>
<th>Status</th>
<th>Conflicts</th>
<th>Current State</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOD_CHECK_IN_PROGRESS</td>
<td>Unknown</td>
<td>Request sent to AACG and waiting for response</td>
</tr>
<tr>
<td>SOD_REMEDIATION_IN_PROGRESS</td>
<td>Conflict found</td>
<td>AACG detected violations and remediation is in progress</td>
</tr>
<tr>
<td>SOD_CHECK_APPROVED</td>
<td>No conflict found</td>
<td>No SOD violations found</td>
</tr>
<tr>
<td>SOD_CHECK_REJECTED</td>
<td>Conflict found</td>
<td>AACG detected violations that cannot be remediated</td>
</tr>
<tr>
<td>SOD_REMEDIATION_APPROVED</td>
<td>Conflict found</td>
<td>AACG detected violations that are approved</td>
</tr>
<tr>
<td>SOD_REMEDIATION_REJECTED</td>
<td>Conflict found</td>
<td>AACG detected violations that are rejected by approver</td>
</tr>
</tbody>
</table>

In the absence of an SOD exception, OIM provisions all relevant users.

**Note**

When a partner user is provisioned, all employees of the partner enterprise are provisioned. SOD checks occur when an external user requests to join a program, because SOD policies operate across Oracle Fusion Applications, not at the individual level. Supplier or partner company user requests are not approved if there is an SOD conflict against the supplier company.

OIM provides AACG with the details of SOD exception approval workflow. AACG audits the outcome for use in future detective controls and audit processes.

**Oracle Application Access Controls Governor**

AACG may respond with the following.
• Roles may be provisioned to the external user or its employees because no SOD conflict is found

• SOD conflict is found and request is denied because the relevant SOD policy is to be strictly enforced and no exception approval should be allowed

• SOD conflict is found and the exception to the policy is allowed, so the request goes through additional processing, such as an approval process.

Supplier or Partner Relationship Management responds to an SOD exception by updating Trading Community Model tables with the current state. An enterprise may elect to implement a landing pad that offers external users a means of addressing the SOD problem by providing more information or withdrawing the request.

SOD violation checking occurs during role implementation and provisioning, and can be turned on or off if AACG is provisioned and enabled as part of the Oracle Fusion Applications deployment.

**Segregation of Duties Exception Resolution or Approval Workflow**

Depending upon status, OIM kicks off an auditable SOD exception resolution workflow. Resolution can be conditional based on approval or requirements such as contracts being met.

If one of the paths for exception resolution is to get an approval, then the SOD exception resolution drives the approval using AMX. Standard AMX rules, not business rules, resolve the approval for the SOD exception, including the following.

• Organizational hierarchies

• Multiple mandatory and optional approvers

• Rerouting and approval delegation

The approver resolution uses AMX Rules Designer to access various user attributes and organizational hierarchies managed in Oracle Fusion Applications repositories. This information is typically not available in OIM or the LDAP identity store repository. Enterprises can define additional approval rules using AMX Thin Client.

The SOD Exception Approver gets a notification through supported channels that a new request is awaiting approval. The approver signs in to the global SOA federated worklist application that aggregates all pending worklist items for the user from all Oracle Fusion applications and logical partitions or pillars of applications. The SOD exception approval tasks show up in the same list.

The SOD exception approval task shows the details of the SPML request and SOD Provisioning results in a page rendered by OIM. The approver may take one of the following actions.

• Approve the request as it is

• Reject the request

If the approver approves the request, OIM sends an SOD_REMEDIATION_APPROVED status to the SPML client.
If the approver rejects the request, OIM sends an SOD_REMEDIATION_REJECTED status to the SPML client. The provisioning request is considered completed with a failure outcome and the external users is notified. Oracle Fusion Applications updates the Trading Community Model tables with the rejected status.

**Remediation Task Assignments**

The SOD remediation tasks are assigned based on the role being requested.

1. If the role requested is Chief Financial Officer, the SOD remediation task is assigned to the IT Security Manager role.

2. If the SOD violation results from a policy where the SOD control tag is the Information Technology Management business process and the control priority is 1, the SOD remediation task is assigned to Application Administrator role.

3. In all other scenarios, the SOD remediation task is assigned to the Controller role.

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

For information on managing segregation of duties, see the Oracle Application Access Controls Governor Implementation Guide and Oracle Application Access Controls Governor User's Guide.
Common Applications Configuration: Define Approval Management for Customer Data Management

Approval Management: Highlights

Use approval management to determine the policies that apply to approval workflows for particular business objects such as expense reports. For example, you can specify levels of approval for expense reports over a particular amount, to reflect your own corporate policies. You also determine the groups of users who act on these workflow tasks, for example, the chain of approvers for expense reports.

Approval management is fully described in the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management. Though the concepts described there apply also to Oracle Fusion Applications, the only setup relevant to Oracle Fusion Applications involves approval groups and task configuration. Customization of approval workflows themselves is described in the Oracle Fusion Applications Extensibility Guide for Developers.

Overview


  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
Common Applications Configuration: Define Help Configuration

Define Help Configuration: Overview

The Define Help Configuration task list contains tasks that let you set up and maintain Oracle Fusion Applications Help for all users. Use the Set Help Options task to determine if certain aspects of Oracle Fusion Applications Help are available to users and to control how aspects of the help site work. Use the Assign Help Text Administration Duty and Manage Help Security Groups tasks to set up customization of help content.

After performing the help configuration tasks, you can review the predefined help and consider whether to add or customize any content. Help that is embedded in the application, for example hints, can also be customized.

Use the Setup and Maintenance work area to access the tasks in the Define Help Configuration task list.

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

Define a help security group and assign a duty role to the group.

1. From the Setup and Maintenance work area, find the Manage Help Security Groups task and click Go to Task.
3. Complete the fields, as shown in this table. Leave the start and end dates blank.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Help Security Group</td>
<td>HR</td>
</tr>
<tr>
<td>Meaning</td>
<td>HR Only</td>
</tr>
<tr>
<td>Description</td>
<td>Viewing by HR specialists only</td>
</tr>
<tr>
<td>Display Sequence</td>
<td>1</td>
</tr>
</tbody>
</table>

4. Click Save.
5. With your new help security group selected, go to the Associated Roles section and add a new row.
6. Select PER_HUMAN_RESOURCE_SPECIALIST as the role name.
7. Click Save and Close.

You have created a new lookup code for the Help Security Groups lookup type, which is a standard lookup. The lookup code has the name, meaning, and description that you defined for the help security group.

You have also created a data security policy for the help database resource, specifying that the Human Resource Specialist role can view help that is defined with the HR security group. If you go to the Manage Database Resources and Policies page and find the database resource, or object, ATK_KR_TOPICS, then you can see the policy for the Human Resource Specialist role.
Resource Specialist role, with the condition that the column name, SECURITY_CODE, is equal to the value HR.
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 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.
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.

Manage Public Unique Identifier Profile Options

Public Unique Identifier: Explained

Oracle Sales Cloud generates thousands of rows of data that require unique identifiers (UIDs). These automatically generated, 18-digit numbers are not
easily read or used by humans. A cogent example is an electronic airline ticket whose numbers can be 13 digits long, such as, 0162128736572. There are also confirmation or itinerary numbers that relate directly to that ticket that are only six digits long, such as QDLG9S; these can be used interchangeably with the ticket number. These shorter confirmation numbers are much easier to read and process by passengers and employees alike. A separate Public Unique ID (PUID) is also generated by the document sequencing feature and is available for encoding into a more user-friendly, alphanumeric ID.

The default setup shows the automatically generated numeric private UIDs and PUIDs for data rows. To use an alphanumeric PUID instead of the delivered numeric, navigate to the Setup and Maintenance task Manage Public Unique Identifier Profile Options to enable the encoding process. There are two profile options to configure the PUID:

- **CRM Public Unique ID String Encoding** controls the characters used in the encoding of the PUID based on a radix, or base number.
- **CRM Public Unique ID Prefix** defines the optional prefix value for the PUID.

Document sequencing typically begins with the number one. You may choose to start the sequences of your PUIDs at a specific value. See the related links below to determine the document sequencing.

**CRM Public Unique ID String Encoding Profile Option**

This profile option determines the set of numbers and letters to be used in encoding the PUID generated by the document sequencing feature. The PUID is delivered with the encoding set to none. To enable the encoding of the PUID, choose a radix, or base conversion algorithm. The available values are stored in the lookup type ZCA_PUID_ENCODING. These encoding methods convert the PUID into user readable IDs using alphanumeric characters rather than just numeric digits. For example, if BASE_26 is chosen, only the upper case letters of the alphabet will be used.

There are seven delivered base values. These are not extensible.

<table>
<thead>
<tr>
<th>RADIX</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE_16</td>
<td>Numbers 0-9, letters A-F</td>
</tr>
<tr>
<td>BASE_26</td>
<td>Letters A-Z, Upper Case</td>
</tr>
<tr>
<td>BASE_29</td>
<td>Numbers 2-9, letters BCDFGHJKLMNPQRSTVWXYZ</td>
</tr>
<tr>
<td>BASE_31</td>
<td>Numbers 0-9, letters ABCDEFGHJKLMNPQRSTUVWX</td>
</tr>
<tr>
<td>BASE_36</td>
<td>Numbers 0-9, letters A-Z upper case</td>
</tr>
<tr>
<td>BASE_62</td>
<td>Numbers 0-9, letters A-Z, letters a-z</td>
</tr>
</tbody>
</table>
CRM Public Unique ID Prefix Profile Option

A prefix may be defined as a site-level profile option. This profile option is delivered with no prefix defined. When defined, the inserted text is prepended to the PUID base encoded document sequence value. For example, if the PUID was CLE123 and the Prefix was ORA, the new PUID would be ORACLE123. Any delimiters or separation characters desired must be defined in the prefix itself. Ensure that the prefix length is short enough so that the concatenated PUID and prefix do not exceed the defined field length, usually no more than 30 characters. For example, you may want the records for the pharmaceutical divisions of your company to be denoted with PUIDs and the prefix Pharma- or Pharma1, Pharma2, and so on.

Set Activity Stream Options

Setting Activity Stream Preferences: Procedures

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 all activity stream regions for all users across your site. Individual users can still override your settings through Activity Stream preferences.

Setting Preferences

Using the activity stream preferences you can specify who can view your activity stream, for which users, services, and spaces to track activities, and the activities to show in an activity stream task flow.

Perform the following steps to set the preferences.

1. In the Setup and Maintenance work area, search for the Set Activity Stream Options task and open it.

2. On the preferences page, click People and select one of the following options:

   Tip

   This setting relates only to the activities that stream from the people connections service. Such activities include making connections, posting feedback and messages, adjusting your profile, and so on.

   • Only Me - to display your own activities in your view of the activity stream.
• Me and My Connections - to display your activities and the activities of your connections in your view of the activity stream.
• No Personal - to hide any user activity in your view of the activity stream, including your own.

3. Click **Spaces** and select one of the following options:
   • All Spaces - to stream activities from all available spaces.
   • My Spaces - to stream activities from the spaces of which you are a member.
   • No Spaces - to avoid streaming any activities from spaces other than the home space.

4. Click **Service Categories** and select the services for which you want to track and display the activities.

**Tip**
If you select No Spaces under Spaces (in the earlier step), the services do not publish any activity to your view of the activity stream, even if you select the services here.

5. Click **Privacy** and select one of the following options:
   • Everyone - all users, whether they are signed in or not, can see your view of the activity stream.
   • Authenticated Users - all users who are signed in can see your view of the activity stream.
   • My Connections - everyone connected to you can see your view of the activity stream.
   • Myself - only you can see your view of the activity stream.

6. Click **Comments and Likes** and select the required options.

7. Click **Save**.

**Manage Menu Customizations**

**Menu Customization: Explained**

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

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

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

**Tip**

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

**Navigator Menu Configuration**

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

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

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

**Note**

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

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

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

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

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

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

Adding Navigator Menu Group: Points to Consider

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

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

Adding Groups

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

Adding Menu Items: Points to Consider

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

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

Adding a Home Page Menu Item

To add a home page menu item, navigate to the place where you want the item to appear and insert it above or below the existing item. You can also duplicate
an existing menu item and position it at the required location. You must provide a label for the menu and link the menu item to a UI Shell page.

Adding a Navigator Menu Item

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

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

You can link a Navigator menu item to the following:

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

Linking to a UI Shell Page

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

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

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

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

For non-Cloud applications, you can determine the secured resource name by obtaining the name of the web page’s page definition file. By default, the page definition files are located in the view.PageDefs package in the Application Sources directory of the view project. If the corresponding JavaServer Faces (JSF) page is saved to a directory other than the default (public_html), or to a subdirectory of the default, then the page definition will also be saved to a package of the same name. An example of a secured resource name is oracle.apps.View.pageDefs.CaseList_Form_Attach_UIShellPagePageDef.

A UI Shell page might take parameters and display or act differently based on the parameters that are passed in. For example, if accessing a page from
one group in the menu hierarchy, the parameter might be set to \texttt{status=Open}
and if accessing the page from a different group, the parameter might be set to  
\texttt{status=Closed}. If the page takes parameters, you can use the Page Parameters
List text box to provide a semicolon-delimited string of name-value pairs, such
as \texttt{org=m1;context=s1}. You can use expression language (EL) to specify the
parameters. If the EL evaluates to an Object, the \texttt{toString} value of that Object is
passed as the value of the parameter.

\section*{Linking to the Dynamic URL of an External Web Site or Application}

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

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

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

When the complete URL to be linked is: \texttt{http://example:9011/myApp/faces/}
\texttt{Page1},

- The name of the web application added to topology would be: \texttt{myApp}
  (the value that would eventually appear in the Web Application list) and
  the protocol host, port, and context root values of the URL would be:
  \texttt{http://example:9011/myApp}

- The value to be provided in the Destination for Web Application field
  would be: \texttt{/faces/Page1}

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

\section*{Linking to a Static URL of an External Web Site or Application}

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

\section*{Hiding or Displaying Menu Nodes: Points to Consider}

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

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

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

If you want a menu group or a menu item to appear only if certain conditions are met, you can use an expression language (EL) command to make the node to appear. For example, #{securityContext.userInRole['ADMIN']}. A node that appears in italics either contains an EL command or the Rendered check box beside it was cleared, and therefore is hidden from end users.

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

Design Time Menu Customizations: Highlights

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

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

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

Customizations

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

• Define translations for your customizations in the locales you support.
  See: Translating Menu Customizations

• Customize the page template to display the Navigator menu groups as separate menus, each of them displaying their list of menu items. Refer to the Oracle Fusion Applications Developer’s Guide.
  See: Rendering the Navigator Menu as Dropdown Buttons

Troubleshooting Navigator Menu: Highlights

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

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

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

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

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

Manage Audit Policies

Managing Audit Policies: Explained

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

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

Enabling Audit Functionality

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

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

Setting Up Functional Security for Audit: Worked Example

Functional security for access to the central audit reporting UI and the setup UI requires manual configuration in Authorization Policy Manager (APM). This topic explains how to perform the necessary tasks for this configuration.

The tasks to configure access to the audit setup UI are:

- Create the Duty Role to Manage Audit History
- Create the Policy to Manage Audit History
- Map Job Roles to Manage Audit History

The tasks to configure access to the audit reporting UI are:

- Create the Duty Role to View Audit History
- Create the Policy to View Audit History
- Map Job Roles to View Audit History

Each task is described in the following sections.

Note
By default, the IT Security Manager job role has access to APM. You must have this job role or another with APM access in order to perform these steps.

Creating the Duty Role to Manage Audit History

1. Open the APM application using one of the following methods:
   - Enter the appropriate URL for your environment.
   - Navigate to Setup and Maintenance. On the All Tasks tab, select "Task Lists and Tasks," search for the "Manage Data Security Policies" task, then click the Go To Task icon in the search results.

2. On the Home tab, select fsdm as the application name.

3. Click New under Application Roles.

4. As the Display Name, enter Manage Audit History.

5. As the Role Name, enter MANAGE_AUDIT_HISTORY.

6. Click Save on the tab for the newly created role.

Creating the Policy to Manage Audit History

1. On the Manage Audit History tab, click Create Policy and select the default policy domain.
2. As the Display Name, enter Manage Audit History.

3. As the Policy Name, enter MANAGE_AUDIT_HISTORY.

4. Click the plus sign to add a new Target.

5. Search for a Display Name beginning with Manage Audit Policies. The search should return one result: FND_MANAGE_AUDIT_POLICIES_PRIV.

6. Click Add All.
7. Click **Add Targets** at the bottom of the page.

8. Click **Save** on the tab for the newly created policy.

**Map Job Roles to Manage Audit History**

1. Click the Home tab.
2. Select fscm as the application name if it is not already selected.
3. Click **Search** under Application Roles.
4. Search for a Display Name beginning with Manage Audit History. The search should return one result: MANAGE_AUDIT_HISTORY.

5. Click Open, then click the External Role Mapping tab.

6. Click the plus sign to add an external role mapping.

7. Search for the job roles that should have access to the audit setup UI, and click Map Roles. If there is an inheritance at the duty role level, but the
corresponding job roles do not inherit from each other, you must map both job roles here.

Create a Duty Role to View Audit History

1. Click the Home tab.
2. Select fscm as the application name if it is not already selected.
3. Click **New** under Application Roles.
4. As the Display Name, enter View Audit History.
5. As the Role Name, enter VIEW_AUDIT_HISTORY.
6. Click **Save** on the tab for the newly created role.

**Create a Policy to View Audit History**

1. On the View Audit History tab, click **Create Policy** and select the default policy domain.
2. As the Display Name, enter View Audit History.
3. As the Policy Name, enter VIEW_AUDIT_HISTORY.
4. Click the plus sign to add a new Target.

5. Search for a Display Name beginning with View Audit History. The search should return one result: FND_VIEW_AUDIT_HISTORY_PRIV.
6. Click Add All.
7. Click **Add Targets** at the bottom of the page.

8. Click **Save** on the tab for the newly created policy.

**Map Job Roles to View Audit History**

1. Click the Home tab.

2. Select fscm as the application name if it is not already selected.

3. Click **Search** under Application Roles.
4. Search for a Display Name beginning with View Audit History. The search should return one result: VIEW_AUDIT_HISTORY.

5. Click Open, then click the External Role Mapping tab.

6. Click the plus sign to add an external role mapping.

7. Search for the job roles that should have access to the audit reporting UI, and click Map Roles. If there is inheritance at the duty role level, but the
corresponding job roles do not inherit from each other, then you must map both job roles here.

FAQs for Manage Audit Policies after Manage Audit Policies

What Oracle Customer Experience Cloud objects can I enable to track their audit history?

Auditing is used to monitor user activity and all configuration, security, and data changes that have been made to an application. You can enable business objects to allow auditing, recording and retrieving information about their creation, modification, and removal.

You can enable the following objects:

<table>
<thead>
<tr>
<th>Area</th>
<th>Parent Objects</th>
<th>Child Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opportunities</td>
<td>Opportunity</td>
<td>Opportunity Revenue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opportunity Split Revenue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opportunity Recurring Revenue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opportunity Revenue Line Set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opportunity Team Member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opportunity Partner</td>
</tr>
<tr>
<td>Marketing</td>
<td>Leads</td>
<td>Leads Product</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Leads Resources</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Leads Contacts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Leads Territories</td>
</tr>
<tr>
<td>Area</td>
<td>Parent Objects</td>
<td>Child Objects</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>----------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>Marketing</td>
<td>Budget</td>
<td>Budget Entries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fund Requests</td>
</tr>
<tr>
<td>Partner Relationship Management (PRM)</td>
<td>Partner</td>
<td>Classification Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Partner Type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Customer Contact Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Partner Certification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Expertise</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Geographies Served</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Industries Served</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Product Specialties</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Account Team</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E-Mail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instant Messaging</td>
</tr>
<tr>
<td>Partner Relationship Management (PRM)</td>
<td>Partner Program</td>
<td>Program Benefit Details</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Program Objective Details</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Program Contract Templates</td>
</tr>
<tr>
<td>Partner Relationship Management</td>
<td>Partner Enrollment</td>
<td>Enrollment Participants</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Contract Enrollments</td>
</tr>
<tr>
<td>Area</td>
<td>Parent Objects</td>
<td>Child Objects</td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>---------------------------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>Trading Community Architecture (TCA)</td>
<td>Account (Account Profile)</td>
<td>Sales Account Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Territory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address Purpose</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E-Mail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instant Messaging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Classification Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Account Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Contact Preference</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Usage Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Identifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Source System References</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Relationship</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Customer Contact Profile</td>
</tr>
<tr>
<td>Area</td>
<td>Parent Objects</td>
<td>Child Objects</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>Trading Community Architecture</td>
<td>Contact (Person Profile)</td>
<td>Sales Account Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Territory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Contact Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address Purpose</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E-Mail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instant Messaging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Classification Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Contact Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Contact Preference</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Usage Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Identifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Source System References</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Relationship</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Customer Contact Profile</td>
</tr>
<tr>
<td>Area</td>
<td>Parent Objects</td>
<td>Child Objects</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Trading Community Architecture</td>
<td>Household (Household Profile)</td>
<td>Sales Account Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Account Territory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address Purpose</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E-Mail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instant Messaging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Classification Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Household Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Contact Preference</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Usage Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional Identifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Source System References</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Relationship</td>
</tr>
<tr>
<td>Trading Community Architecture</td>
<td>Resource (Resource Profile)</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address Purpose</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Location SDO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E-Mail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instant Messaging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource Organization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Membership</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource Role Assignment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sales Representative</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource Team Membership</td>
</tr>
</tbody>
</table>
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.

Note

Once shared, all updates to the enabled attributes of the object instance, and deletes, are sent to social network. Updates to attributes that are not enabled are not sent.
• **Automatic**: Automatically sends the newly enabled object instances and updates to social network.

**Note**

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

• **No**: Does not send any information on object instance to social network. This is the default option.

3. Click **OK**.

This enables the selected business object, and empowers the user to decide whether to share each instance of the object with social network.

**Note**

After you enable an object, you must enable one or more attributes in the **Attributes** section of the **Manage Oracle Social Network Objects** page. Updates to enabled attributes are sent to social network.

### Disable Object

To disable a business object:

1. Access the **Manage Oracle Social Network Objects** page.

2. In the **Business Objects** section, select a business object, and click **Disable Object**.

3. Save your changes.

This disables the selected business object by updating the enablement option as **No**.

### Enable All

To enable all business objects:

1. Access the **Manage Oracle Social Network Objects** page.

2. In the **Business Objects** section, click **Enable All**.

3. Save your changes.

This enables all business objects in bulk, and updates the enablement option of all business objects as **Manual**.

**Note**

• After you enable business objects, you must enable one or more attributes in the **Attributes** section of the **Manage Oracle Social Network Objects** page. Updates to the enabled attributes are sent to social network.

• If you enable a business object, but don’t configure any attributes for the enabled business object, no attributes are sent to social network during
create and update. The only exception is that some internal bookkeeping information are sent. Deletes are sent as usual.

**Status Column**

The Status column in the **Business Objects** table visually indicates:

- Whether a business object is enabled
- Which enabled business objects don’t yet have an enabled attribute assigned

The status indicators include:

- A check mark, which indicates that you have configured attributes for an enabled business object
- A warning sign, which indicates that you have not configured any attributes for an enabled business object

**Enable Business Object Attributes**

To enable business object attributes:

1. In the **Attributes** section, click **Add** to display the **Select Attributes** dialog where you can select attributes to add to the table.
2. Select an attribute name in the table, and select the **Enabled** check box to enable the attribute.
3. Click **OK**.
4. Save your changes.

In the **Attributes** table, you can also:

- Click **View** to view a list of all attributes that are enabled.
- Click **Remove** to remove attributes from the table.
- Hover over the **Attribute Information** icon displayed next to descriptive flexfield attributes to view information about the attributes.

**Update Translations: Explained**

The Update Translations process sends attribute labels and business object names to Oracle Social Network Cloud Service for use in the user interface.

In social network, the attribute or business object labels appear in the language of your locale. If you change the locale in social network, then the attribute or business object labels appear in the updated language. However, the data appears in the language in which it was originally sent to social network. If you have previously sent an instance of the business object to social network, then the instance data isn’t updated. Clicking the **Update Translations** button on the **Manage Oracle Social Network Objects** page sends translations for business objects with the enablement option as **Manual** or **Automatic**.
Synchronize Business Objects: Explained

Use the Synchronize button on the Manage Oracle Social Network Objects page to synchronize business objects. This re-sends the definitions of business objects having the enablement option as Manual or Automatic to Oracle Social Network Cloud Service.

Use the Synchronize button at the:

- **Business Objects table level**: To re-send the definitions of a selected business object to social network. This button is enabled only when you select a row for a business object with the enablement option as Manual or Automatic.

- **Manage Oracle Social Network Objects page level**: To re-send the definitions of all business objects with the enablement option as Manual or Automatic to social network.

**Note**

If you had modified any business object enabled for social network and not saved your changes, then on clicking the Synchronize button, a warning message appears. This message informs you that you have not saved your changes, and you can select one of the following options:

- **Save and Synchronize**: To save the modified business objects, and synchronize the unmodified business objects.

- **Synchronize**: To ignore any unsaved business objects, and only synchronize the unmodified business objects.

- **Cancel**: To cancel the synchronization task.

FAQs for Manage Oracle Social Network Objects

**What happens if I update translations?**

When you update translations, you send translations for business objects with the enablement option as Manual or Automatic to Oracle Social Network Cloud Service.

On updating translations, you also:

- Synchronize the newly translated text from Oracle Fusion Applications so that it can be used within social network. This means you can:

  - Install and enable a new language.
  - Take a language patch at any time.

  - Send attribute labels and business object names to social network for use in its user interface.
How can I update translations?

Use the **Update Translations** button on the **Manage Oracle Social Network Objects** page for subsequent updates to labels and attributes.

Use the **Update Translations** button at the:

- **Business Objects table level**: To send translations for a selected business object to Oracle Social Network Cloud Service. This button is enabled only when you select a row for a business object with the enablement option as Manual or Automatic.

- **Manage Oracle Social Network Objects page level**: To send translations for all business objects with the enablement option as Manual or Automatic to social network.

**Note**

When you save the enablement of a business object to social network, it sends the translations as well. Hence, you need not click the **Update Translations** button after saving the enablement.

When do I update translations?

Run the **Update Translations** process only after you install a new language pack of Oracle Fusion Applications.

Updating translations synchronizes the newly translated text to Oracle Social Network Cloud Service for integration with Oracle Fusion Applications.

**Note**

When you save the enablement of a business object to social network, it sends the translations as well. Hence, you need not click the **Update Translations** button after saving the enablement.

What happens if I synchronize business objects?

When you synchronize business objects, you re-send the definitions of business objects having the enablement option as Manual or Automatic to Oracle Social Network Cloud Service.

When do I synchronize business objects?

Run the Synchronize process after you use customization sets to import the setup from the **Manage Oracle Social Network Objects** page in another environment.
You can also run the process any time you want to synchronize the settings of business objects with Oracle Social Network Cloud Service without making any changes in the Manage Oracle Social Network objects page.

Manage Applications Core Common Reference Objects

Manage Applications Core Messages

Common Messages: Points to Consider

Common messages, which have message names that begin with FND_CMN and message numbers between 0 and 999, are used throughout Oracle Fusion Applications. Each common message can appear in multiple places in any product family. For example, the FND_CMN_NEW_SRCH message can be used for any search to indicate that no results were found. Common messages that are of type error or warning are part of the message dictionary.

Editing Common Messages

Because a common message can be used in any application, consider the ramifications if you edit any aspect of the message, including incident and logging settings. Changes would be reflected in all instances where the message is used. For example, if you change the message text, make sure that the text would make sense to all users across Oracle Fusion Applications who might see it.

Creating Common Messages

You can create custom common messages for use in multiple places within a single product. Do not begin the message name with FND_CMN, but use another suitable convention. The message number should be within the range that is designated for the product.

Manage Applications Core Administrator Profile Values

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

**Profile Options and Related General Preferences: How They Work Together**

Some Oracle Middleware Extensions for Applications profile options are related to general preferences in the global area.

**Preferences**

The related general preferences are Default Application Language, Territory, Date Format, Time Format, Currency, and Time Zone. When the user changes any of these preferences, the stored values in LDAP are updated accordingly.

**Profile Options**

The corresponding profile options are Default Language, Default Territory, Default Date Format, Default Time Format, Default Currency, and Default User
Time Zone. No matter what you set for these profile options at any level, the preferences settings, or LDAP values, take precedence. The profile option value is used only if the LDAP value is not available. Updating the profile option value does not automatically update the value in LDAP or preferences.

FAQs for Manage Applications Core Administrator Profile Values

How can I enable the privacy statement?

Use the Privacy Statement URL profile option to enable the Privacy Statement menu item in the global area. This menu item in the Settings and Actions menu is disabled by default.

Open the Setup and Maintenance work area, and use the Manage Applications Core Administrator Profile Values task to find the Privacy Statement URL profile option. In the Profile Value column, enter the full URL of the web page that contains the privacy content you want the menu item to link to.
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>CallingPartyNoReachable</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 HzCTDPsnGatewayApp 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 HzCTDPsnGatewayApp 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>HzCTDPsnGatewayApp</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
Getting Started with Oracle Fusion CRM Collaboration: Overview

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

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

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

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

Activity Streams

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

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

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

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

**Blogs**

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

**Discussion Forums**

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

Discussions has been enabled for:

- Opportunity
- Customer
- References
- Competitors

**Group Spaces**

Teams of employees can use group spaces to collaborate in the context of an opportunity or customer record. Within a group space, employees can collaborate by engaging in existing discussions or creating new ones, and they can create new wikis. They can also create new group spaces and leverage available features, such as discussions, blogs, wikis, announcements, and so on. By using group spaces, employee groups can contend with work blockages, such as:

- Unstructured sales activities that are not captured.
- Information that is locked in silos.
• Hard-to-find successful or useful content.
• Numerous e-mail threads used for collective content authoring.

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

**Tags**

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

**Wikis**

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

Wikis have been enabled for:
• References
• Competitors

**Group Spaces in Oracle Fusion CRM: Highlights**

Group spaces bring people together in a virtual environment for ongoing interaction and information sharing, enabling the formation and support of social networks. Teams can use group spaces to collaborate through discussions, in the context of an opportunity or customer record, and create new wikis.

Users can also create a new group space and leverage available features, such as discussions, blogs, wikis, announcements, and so on. Group spaces support the collaboration of project teams and communities of interest by providing a dedicated and readily accessible area for relevant services, pages, and content, and by supporting the inclusion of specified members. A member can invite you to join a space, in which case you receive a notification. To see all the spaces that you can join, use the **Spaces** link in the global area or go to the **Spaces** dashboard, which contains a list of the spaces to which you belong, a list of any recommended spaces, and a tool to search for all public spaces that you can join.

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

**Accessing Group Spaces**

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

- Select and add a group space: You can select an existing group space from the list. The application allows all users who have access to the opportunity record to use the features exposed in the selected group spaces.

- Create and link a group space: You can create new group spaces using existing group spaces templates. Users who have access to the opportunity record can use the features that are exposed in the newly created group spaces.

**Setting Up Group Spaces**

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

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

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

**Security for Enabling or Viewing Group Space Widgets**

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

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

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

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

**Optional Enabling**

Deploying organizations have a choice of enabling embedded forums and wikis for sales objects by using two profile options:
• Key: (<Object> Group Space Enabled) Determines whether or not a particular object, such as an opportunity, has support for embedding forums and wikis.

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

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

Embedding Group Space Widgets

To embed group space widgets, use the following actions:

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

• To hide The Discussions tab and Wikis tab with existing embedded content, uncheck the Discussion and Wikis check box.

Group Spaces and Oracle WebCenter

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

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

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

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

Collaboration Capabilities in Oracle Fusion CRM: Highlights

Oracle Fusion CRM Collaboration includes the following features.

Fusion Welcome Page

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

• Opportunities and customer records that a user is following.
• Discussion groups of which the user is a member.
• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available in a group space depend on the template that was used when the group space was created.
• Connections updates.
• Status updates.
• Healthcare management activity stream messages.
• Projects.
• All corresponding comments and likes for any of the above activity messages.

Sales Dashboard Page
The Sales Dashboard shows activity stream messages for the following functions:
• Opportunities and customer records that a user is following.
• Discussion groups of which the user is a member.
• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available on a group space depend on the template that was used when the group space was created.
• Connections updates.
• Status updates.
• All corresponding comments and likes for any of the above activity messages.

Customer Center Overview Page
The Customer Center Dashboard shows activity stream messages for the following functions:
• Opportunities and customer records that a user is following.
• Discussion groups of which the user is a member.
• Updates to wikis, blogs, announcements, and so on, from the group space of which the user is a member. The features available on a group space depend on the template that was used when the group space was created.
• Connections updates.
• Status updates.
• All corresponding comments and likes for any of the above activity messages.

Posting Messages
You can post a status update message from the Oracle Fusion Welcome page. This message can be posted to Everyone, to members of a specific group space, or to members of a specific opportunity or customer sales team. A user can also post a message from an Oracle WebCenter group space. Within a group space, the same field is available for users to post a status update.
• For more information on Oracle Fusion CRM Collaboration capabilities,
Blogs in Oracle Fusion CRM: Highlights

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

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

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

Setting Activity Stream Preferences: Procedures

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 all activity stream regions for all users across your site. Individual users can still override your settings through Activity Stream preferences.

Setting Preferences

Using the activity stream preferences you can specify who can view your activity stream, for which users, services, and spaces to track activities, and the activities to show in an activity stream task flow.

Perform the following steps to set the preferences.

1. In the Setup and Maintenance work area, search for the Set Activity Stream Options task and open it.
2. On the preferences page, click People and select one of the following options:

Tip
This setting relates only to the activities that stream from the people connections service. Such activities include making connections, posting feedback and messages, adjusting your profile, and so on.

- Only Me - to display your own activities in your view of the activity stream.
- Me and My Connections - to display your activities and the activities of your connections in your view of the activity stream.
- No Personal - to hide any user activity in your view of the activity stream, including your own.

3. Click **Spaces** and select one of the following options:

- All Spaces - to stream activities from all available spaces.
- My Spaces - to stream activities from the spaces of which you are a member.
- No Spaces - to avoid streaming any activities from spaces other than the home space.

4. Click **Service Categories** and select the services for which you want to track and display the activities.

**Tip**

If you select No Spaces under Spaces (in the earlier step), the services do not publish any activity to your view of the activity stream, even if you select the services here.

5. Click **Privacy** and select one of the following options:

- Everyone - all users, whether they are signed in or not, can see your view of the activity stream.
- Authenticated Users - all users who are signed in can see your view of the activity stream.
- My Connections - everyone connected to you can see your view of the activity stream.
- Myself - only you can see your view of the activity stream.

6. Click **Comments and Likes** and select the required options.

7. Click **Save**.

**Tags in Oracle Fusion CRM: Highlights**

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

The tags feature is an Oracle WebCenter service.

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

- Disregard discussions specific to the Oracle WebCenter Spaces application, for example the global search in the application.
- Although the guide describes tagging Oracle WebCenter pages and documents, in Oracle Fusion Applications you tag specific business objects.
- Not all tag features are available in Oracle Fusion Applications, for example the Tags and Similarly Tagged Items user interfaces.

**Understanding Tags**

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

- Use the tag icon to tag specific business objects. Read only about the step that describes the fields you enter to tag an item.
- Browse and search for items using tags in the Tag Center dialog box. Disregard instructions about how to open the tag center.
- For more information about tags,
  - See: "Understanding the Tag Center" in Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces
  - See: "Working with Tags and Tagged Items in the Tag Center" in Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces
  - See: "Tagging Application Pages" in Oracle Fusion Middleware User's Guide for Oracle WebCenter Spaces

**Wikis in Oracle Sales Cloud: Highlights**

A wiki is a Web site of interlinked Web pages that allows members of a group to create and edit the Web pages in a Web browser. Members of a group who have the appropriate permissions can add and edit information and share it among the group. Wikis are available by way of Sales Cloud References and Competitors.
In Oracle WebCenter Spaces, you can create and manage wiki documents by using the Documents service. To support the wiki functionality, the Documents service stores all wiki documents on Oracle Content Server. For this feature to work, the Oracle Content Server needs to be set up.

**Setting Up Wikis**

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

- In the Edit Opportunity page, to enable a wiki for an opportunity, click an Opportunity Name.
- Then in the Edit Opportunity page Summary region, check the Show Discussions and Wiki check box and then save.
- This option creates wikis for the opportunity and a Wiki tab about competitors. When a wiki is enabled on the opportunity, anyone with a minimum of view access on the opportunity can contribute to it.
- After you enable wikis, you cannot disable them.

**See Also**

- For more information about wikis, see the following document.

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

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

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

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

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

- In Oracle Fusion Applications, discussions can exist outside of spaces. Content that is specific to spaces applies to spaces within Oracle Fusion Applications, but discussions work similarly outside of spaces.
- Various user interfaces, or task flows, are described in the guide. The interfaces and flows that are available to you depend on the pages that you can access.
## Setting Up and Using Discussion Forums

Moderators can create discussion forums in which members can view and participate in topics and threads. Set up and use discussion forums in the following ways:

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

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

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

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

### See Also

- For more information about discussion groups,

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

## Oracle Sales Cloud Social Network Administration: Overview

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

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

You can integrate Oracle Sales Cloud with Oracle Social Network by enabling end users to create Oracle Social Network Social Objects based on object records.
in Oracle Sales Cloud. By default, Oracle Sales Cloud objects are not enabled for Oracle Social Network. This topic explains how to enable objects so that users can create Oracle Social Network Social Objects based on object records, and view object-related status updates.

The following table shows the Oracle Sales Cloud objects you can enable for Oracle Social Network:

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

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

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

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

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

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

To enable an Oracle Fusion object for Conversations:

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

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

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

To enable individual attributes:

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

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

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

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

Oracle Sales Cloud Social Network Administration: Overview

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

Enabling Objects for Oracle Social Network

Oracle Social Network supports the integration of a number of objects with Oracle Sales Cloud. This includes support for:

- Custom fields on delivered objects
- The ability to share custom objects to Oracle Social Network.

You can enable the following objects for Oracle Social Network:

- Competitor
- Contact
- Group
- Lead
- Opportunity
- Organization
- Partner Account
- Person
- Reference Customer
- Sales Account

The following field types are supported:

- Text
- Number
- Date
- Percentage
- Datetime
- Currency
- Fixed and Dynamic choice lists

Additionally, the following columns of custom objects are supported:

- Created By and Created Date
- Updated By and Updated Date
- RecordName
You can enable entire objects or only certain attributes. When an object is enabled, you can set it to either Automatic or Manual sharing. Automatic sharing means that the Social Object in Oracle Social Network that corresponds to the Oracle Sales Cloud business object is created automatically when the object record is created. If an object is set to Manual sharing, the Oracle Sales Cloud Social Object is not created automatically when the object record is created.

To enable an Oracle Sales Cloud object for Oracle Social Network:

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

A dialog box displays where you can specify whether to set the object to Automatic or Manual sharing, or to turn off sharing entirely. You can also select individual attributes on an object to include in Social Objects.

To enable individual attributes:

1. After you have selected an object on the Manage Social Network Objects tab, expand the Attributes section for the object.
2. Click the New icon.
3. Choose the attributes you want to include by checking their Enabled check boxes.
   
   You must select at least one attribute for an object.
4. Click Save.
   
   A green check mark will appear after you’ve selected an attribute; this validates that the object was set up correctly.

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

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

Removing the Oracle Social Network Conversation Tab and Tree Node in Oracle Sales Cloud: Worked Example

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

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

**Note**

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

**Removing the Conversations Subtab for Opportunity**

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

2. From the Navigator, navigate to Opportunities.


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

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

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

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


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

12. Click OK.

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

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

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

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

5. Select the Site level and click OK.

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

8. Highlight Conversations in the Tree Nodes list.

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

FAQs for Define Social Networking

How can I enable social networking features?

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

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

Note

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

Define Applications Core Configuration: Overview

The Define Applications Core Configurations task list contains the Oracle Middleware Extensions for Applications (Applications Core) tasks that support implementation of common functionality such as lookups, profile options, document sequences, and so on.

Use this task list to manage configuration objects that are defined centrally and shared across applications, in addition to those that are classified under the Maintain Common Reference Objects task list. You can search for this task list in the Setup and Maintenance work area.

Define Lookups

Lookups: Explained

Lookups are lists of values in applications. You define a list of values as a lookup type consisting of a set of lookup codes, each code’s translated meaning, and optionally a tag. End users see the list of translated meanings as the available values for an object.

Lookups provide a means of validation and lists of values where valid values appear on a list with no duplicate values. For example, an application might store the values Y and N in a column in a table, but when displaying those values in the user interface, Yes or No (or their translated equivalents) should be available for end users to select. For example, the two lookup codes Y and N are defined in the REQUIRED_INDICATOR lookup type.

In another example, a lookup type for marital status has lookup codes for users to specify married, single, or available legal partnerships.

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Lookup Code</th>
<th>Meaning</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAR_STATUS</td>
<td>M</td>
<td>Married</td>
<td></td>
</tr>
</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>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.

<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.

**Note**

You can only create or edit the lookup codes for a particular lookup type if its customization level supports it.

**Creating a Lookup Type Called COLORS**

Your enterprise needs a list of values for status to be used on various objects such as processes or users. The lookups are colors, so the lookup type you create is COLORS.

<table>
<thead>
<tr>
<th>Lookup type parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup type name</td>
<td>COLORS</td>
</tr>
<tr>
<td>Meaning</td>
<td>Status</td>
</tr>
<tr>
<td>Description</td>
<td>Status by color</td>
</tr>
<tr>
<td>Module</td>
<td>Oracle Middleware Extensions for Applications</td>
</tr>
</tbody>
</table>
The lookup codes you define for the COLORS lookup type are, BLUE, RED, GREEN, and YELLOW.

<table>
<thead>
<tr>
<th>Lookup Code</th>
<th>Meaning</th>
<th>Enabled</th>
<th>Display Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE</td>
<td>Urgent</td>
<td>No</td>
<td>4</td>
</tr>
<tr>
<td>RED</td>
<td>Stop</td>
<td>Yes</td>
<td>1</td>
</tr>
<tr>
<td>GREEN</td>
<td>Go</td>
<td>Yes</td>
<td>3</td>
</tr>
<tr>
<td>YELLOW</td>
<td>Caution</td>
<td>Yes</td>
<td>2</td>
</tr>
</tbody>
</table>

### Understanding the Resulting Data Entry List of Values

Users need to respond to a process question by indicating whether to stop it, use caution, go ahead, or complete it urgently.

The list of values for the COLORS lookup type includes the meanings for the enabled codes.

<table>
<thead>
<tr>
<th>Displayed Value</th>
<th>Hidden ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop</td>
<td>RED</td>
</tr>
<tr>
<td>Caution</td>
<td>YELLOW</td>
</tr>
<tr>
<td>Go</td>
<td>GREEN</td>
</tr>
</tbody>
</table>

### Analysis

The BLUE lookup code was not enabled and does not appear in the list of values. The display sequence of values in the list of values is alphabetical unless you enter a number manually to determine the order of appearance. Number 1 indicates the value listed first in the list of values.

### Note

Only lookups that are enabled and active, meaning between start and end dates, are visible.

### Understanding the Transaction Table

When users enter one of the values from the list of values for the lookup type COLORS, the transaction table records the lookup code. In this example, the code is stored in the Status column.

<table>
<thead>
<tr>
<th>Transaction number</th>
<th>User name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jane</td>
<td>RED</td>
</tr>
<tr>
<td>Transaction number</td>
<td>User name</td>
<td>Status</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------</td>
<td>--------</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 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 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.

**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 Customer.

**Customization**

You can customize the predefined tree structures as well as the ones that you created. However, customizing the predefined tree structures involves certain level of access restrictions, and will be limited to specific tree nodes and downwards in hierarchy.

**Multiple Tree Versions**

One or more trees and tree versions can be based on a tree structure. A tree structure can have one or more trees and tree versions based on it. Usually, only one active version is permitted at any given point of time. However, depending on the requirement, you can allow two or more tree versions to be in the active state for the same date range. This flexibility allows you to choose the tree version that you want to implement.
Managing Tree Structures: Points to Consider

You can create, edit, and delete tree structures depending upon the requirement. You can also audit and change the status a tree structure.

Creating and Editing Tree Structures

You can create trees on the basis of a tree structure. When you edit an active tree structure, the status of the tree structure and all associated trees and their versions change to draft. To reuse a tree structure, you can create a copy of it without copying the associated trees and tree versions. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.

Note
For specific information on working with the predefined tree structures that exist in an Oracle Fusion application, refer to the specific product documentation.

Setting Status

If you change the status of a tree structure, the status of the trees and tree versions associated with that tree structure also changes.

The following table lists the different statuses of a tree structure.

<table>
<thead>
<tr>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>Yet to be published or is in a modified state.</td>
</tr>
<tr>
<td>Active</td>
<td>In use and based on which one or more trees or tree versions are created.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Not in use.</td>
</tr>
</tbody>
</table>

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</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></td>
<td>to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have a reference data set attribute. This validation does not take place when the check box is not selected.</td>
<td></td>
<td></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 FND_TREE_NODE_RF or a custom table.</td>
<td>The specified table does not exist in the database. The specified table does not contain the same columns as the FND_TREE_NODE_RF table.</td>
<td>Correct the row flattened table definition.</td>
</tr>
<tr>
<td>Available Label Data Sources</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 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.</td>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td>Validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------</td>
<td>-------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Available Data Sources</td>
<td>Each data source view object specified for the tree structure must be accessible, and all its primary key attributes must be valid.</td>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td>Column Flattened Table Name</td>
<td>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.</td>
<td>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.</td>
<td>Correct the column flattened table definition.</td>
</tr>
<tr>
<td>Validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------</td>
<td>--------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Restrict by Date</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have effective start date and effective end date attributes. This validation does not take place when the check box is not selected.</td>
<td>Even when the check box is selected, one or more of its data source view objects does not contain effective start date and effective end date attributes.</td>
<td>If the date restriction is required for this tree structure, include the effective start date and effective end date attributes on all data sources. Otherwise, deselect the check box.</td>
</tr>
<tr>
<td>Tree Node Table Name</td>
<td>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.</td>
<td>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.</td>
<td>Correct the tree node table definition.</td>
</tr>
<tr>
<td>Allow Node Level Security</td>
<td>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.</td>
<td>The option is set to No for the tree structure but one or more associated data sources have that option set to Yes.</td>
<td>Correct the option setting in the tree structure and their data sources.</td>
</tr>
</tbody>
</table>

**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 customers assignment. You may also choose not to have any labeling scheme. However, if you decide to use any of the labeling schemes, you may need to select the following additional options, to restrict the list of values that appear under the selected tree node.

- **Allow Ragged Nodes**: To include nodes that have no child nodes, and are shorter than the remaining nodes in the entire hierarchy.
- **Allow Skip Level Nodes**: To include nodes that are at the same level but have parent nodes at different levels.

### Restriction of Tree Node Values

You can decide the depth of the tree structure by selecting an appropriate value from the list. Keeping the depth limit open renders an infinite list of values. Using the following options, you can restrict the list of values that appear for selection under a specific tree node.

- **Date Range**: Specifies whether a selection of nodes should be restricted to the same date range as the tree version.
- **Allow Multiple Root Nodes**: Allows you to add multiple root nodes when creating a tree version.
- **Reference Data Set**: Specifies whether a selection of nodes should be restricted to the same set as the tree.

### Data Source Values and Parameters

Tree data sources have optional data source parameters with defined view criteria and associated bind variables. You can specify view criteria as a data source parameter when creating a tree structure, and edit the parameters when creating a tree. Multiple data sources can be associated with a tree structure and can have well-defined relationships among them.

**Note**

Parameter values customized at the tree level override the default values specified at the tree-structure level.

The data source parameters are applied to any tree version belonging to that data source, when performing node operations on the tree nodes. Data source parameters also provide an additional level of filtering for different tree structures. The tree structure definition supports three data source parameter types.

- **Bound Value**: Captures any fixed value, which is used as part of the view criteria condition.
- **Variable**: Captures and binds a dynamic value that is being used by the data source view object. This value is used by the WHERE condition of the data flow.
- **View Criteria**: Captures the view criteria name, which is applied to the data source view object.

You can also specify which of the data source parameters are mandatory while creating or editing the tree structure.

View objects from the ADF business components are used as data sources. To associate the view object with the tree structure, you can pick the code from ADF business component view objects and provide the fully qualified name of the view object, for example, `oracle.apps.fnd.applcore.trees.model.view.FndLabelVO.`
Specifying Performance Options for a Tree Structure: Points to Consider

Tree structures are heavily loaded with data. As a tree management guideline, use the following settings to improve performance of data rendering and retrieval.

- Row Flattening
- Column Flattening
- Column Flattened Entity Objects
- ADF Business Component View Objects

Row Flattening

Row flattening optimizes parent-child information for run-time performance by storing additional rows in a table for instantly finding all descendants of a parent without initiating a CONNECT BY query. Row flattening eliminates recursive queries, which allows operations to perform across an entire subtree more efficiently.

To store row flattened data for the specific tree structure, users can either use the central FND_TREE_NODE_RF table or they can register their own row flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a row-flattened table contains an additional row with Corporation directly being the parent of Region (Corporation-Region).

Column Flattening

Column flattening optimizes parent-child information for run-time performance by storing an additional column in a table for all parents of a child.

To store column flattened data for the specific tree structure, users can either use the central FND_TREE_NODE_CF table or they can register their own column flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a flattened table in addition to these columns, contains three new columns: Region, Sales Division, and Corporation. Although positioned next to each other, the column Region functions at the lower level and Corporation at the higher level, retaining the data hierarchy.

Column Flattened Entity Objects

In the absence of a column-flattened table, if you need to generate the business component view objects for your tree structure for the flattened table, use the tree management infrastructure to correctly provide the fully qualified name of the entity object for the column flattened table.

ADF Business Component View Objects

View objects from the ADF business components can also be used as data sources, eliminating the need to create new types of data sources. This field
is to store the fully qualified name for the business component view object generated by the tree management for business intelligence reporting and usage. The business component view object is a combination of the tree data source and column flattened entity. Using this option prevents data redundancy and promotes greater reuse of existing data, thereby improving the performance of the tree structure.

**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. In an unbalanced hierarchy, a level may not be equal to depth. In an unbalanced hierarchy, a level may not be equal to depth.</td>
</tr>
</tbody>
</table>

**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>Validator</td>
<td>Description</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>Root Node</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Allow Multiple Root Nodes check box for the Restrict Tree Node List of Values Based on option is not selected, and if the tree structure is not empty, the tree version must contain exactly one root node. This validation does not take place if the check box is selected.</td>
<td>Even if the check box is deselected, the tree version has multiple root nodes.</td>
<td>Modify the tree version such that there is exactly one root node.</td>
</tr>
<tr>
<td>Data Source Max Depth</td>
<td>For each data source in the tree structure, on the Data Source dialog box, if the data source is depth-limited, the data in the tree version must adhere to the specified depth limit. This validation does not apply to data sources for which the Maximum Depth field is set to Unlimited.</td>
<td>The tree version has data at a depth greater than the specified depth limit on one or more data sources.</td>
<td>Modify the tree version such that all nodes are at a depth that complies with the data source depth limit.</td>
</tr>
<tr>
<td>Duplicate Node</td>
<td>On the Data Source dialog box, if the Allow Duplicates check box is not selected, the tree version should not contain more than one node with the same primary key from the data source. If the check box is selected, duplicate nodes are permitted.</td>
<td>Even when the check box is deselected, the tree version contains duplicate nodes.</td>
<td>Remove any duplicate nodes from the tree version.</td>
</tr>
<tr>
<td>Validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Available Node</td>
<td>All nodes in the tree version should be valid and available in the underlying data source.</td>
<td>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.</td>
<td>Remove any orphaned nodes from the tree version. Update tree reference nodes so that they reference existing tree versions.</td>
</tr>
<tr>
<td>Node Relationship</td>
<td>All nodes must adhere to the relationships mandated by the data sources registered in the tree structure.</td>
<td>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.</td>
<td>Modify the tree version such that the nodes adhere to the same parent-child relationships as the data sources.</td>
</tr>
<tr>
<td>Validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SetID Restricted Node</td>
<td>On the Manage Tree Structures: Specify Data sources page, if the Set ID check box is selected to enable the Restrict Tree Node List of Values Based on option for 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.</td>
<td>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.</td>
<td>Modify the tree version such that all nodes in the tree have data sources with reference data set matching that of the tree.</td>
</tr>
<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 Labeling Scheme list box, all nodes should have labels. This restriction does not apply when you select None from the Labeling Scheme 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>_validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Date Restricted Node</td>
<td>On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each node in the underlying data source must have an effective date range same as the effective date range of the tree version. This restriction does not apply if the check box is not selected.</td>
<td>Even when the check box is selected, there are data source nodes that have a date range beyond the tree version’s effective date range. For example, if the tree version is effective from Jan-01-2012 to Dec-31-2012, all nodes in the tree version must be effective from Jan-01-2012 to Dec-31-2012 at a minimum. It is acceptable for the nodes to be effective for a date range that extends partly beyond the tree version’s effective date range (for example, the node data source value is effective from Dec-01-2011 to Mar-31-2013). It is not acceptable if the nodes are effective for none or only a part of the tree version’s effective date range (for example, the node data source value are effective only from Jan-01-2012 to June-30-2012).</td>
<td>Ensure that all nodes in the tree version have effective date range for the effective date range for the tree version.</td>
</tr>
<tr>
<td>Multiple Active Tree Version</td>
<td>On the Manage Tree Structures: Specify Definition page, if the Allow Multiple Active Tree Versions check box is not selected for the tree structure, there should not be more than one active tree version under a tree at any time. This restriction does not apply if the check box is selected.</td>
<td>Even when the check box is not selected, there is more than one active tree version in the tree for the same date range.</td>
<td>Set no more than one tree version to Active within the same date range and set the others to inactive or draft status.</td>
</tr>
<tr>
<td>Validator</td>
<td>Description (what is checked)</td>
<td>Possible Cause for Validation Failure</td>
<td>Suggested Corrective Action</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Range Based Node</td>
<td>On the Data Source dialog box, if the Allow Range Children check box is not selected, range-based nodes are not permitted from that data source. This restriction does not apply if the check box is selected.</td>
<td>Even when the check box is not selected, there are range-based nodes from a data source.</td>
<td>Ensure that any range nodes in your tree version are from a data source that allows range children.</td>
</tr>
<tr>
<td>Terminal Node</td>
<td>On the Data Source dialog box, if the Allow Use as Leaves check box is not selected, values from that data source cannot be added as leaves (terminal nodes) to the tree version. This restriction does not apply if the check box is selected.</td>
<td>Even when the check box is not selected, values from a data source are added as leaf nodes (terminal nodes).</td>
<td>Modify the tree version such that all terminal nodes are from data sources for which this check box is selected.</td>
</tr>
<tr>
<td>Usage Limit</td>
<td>On the Data Source dialog box, if the Use All Values option is selected to set the Usage Limit for the data source, every value in the data source must appear as a node in the tree. This restriction does not apply if None option is selected.</td>
<td>Even if the Use All Values option is selected, there are values in the data source that are not in the tree version.</td>
<td>For each data source value that is not yet available, add nodes to the tree version.</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 updatable.

Profile Option Definition and Configuration

Application developers add new profile options and configure ones that are not to be updated by other users. Application administrators and implementation consultants configure profile options with profile option values that are implemented as updatable.

Profile option definitions consist of the following.

• Profile option name
• Application and module in the application taxonomy
• Profile option values
• Profile options categories
• Profile option levels
• Profile option level hierarchy

Profile options can appear on any user interface page without indication that a profile option is what is being set.

**Profile Option Values**

Some profile options have predefined profile option values.

The Manage Profile Option Values task flow allows an administrator to set updatable profile option values at the available levels, including the user level. You can access the Manage Profile Option Values task starting in the Setup and Maintenance Overview page and searching for profile option tasks. You can set profile option values at different levels: site, product, and user. The following table provides examples.

<table>
<thead>
<tr>
<th>Profile Option Level</th>
<th>Value of the Profile Option Level</th>
<th>Profile Option Value</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Manager1</td>
<td>UK pound sterling</td>
<td>Access to site and all products shows UK pounds sterling in effect</td>
</tr>
<tr>
<td>User</td>
<td>Manager2</td>
<td>US dollar</td>
<td>Access to site and all products shows US dollars in effect</td>
</tr>
<tr>
<td>Product</td>
<td>Financials for EMEA</td>
<td>Euro</td>
<td>Unless superseded by a user level value, Euros in effect for Financials for EMEA applications</td>
</tr>
<tr>
<td>Site</td>
<td>Site</td>
<td>UK pound sterling</td>
<td>UK pounds sterling in effect for all other users and products</td>
</tr>
</tbody>
</table>

Context such as user session or accessed product determines which profile option value is associated with the profile option name. In the example, if manager1 does not set a profile option value for this profile option, access to Financials for EMEA shows currency in Euros; and access to other products shows currency in UK pounds sterling.

**Profile Option Categories**

Categories group profile options based on their functional area. Profile option categories facilitate searching and defining data security.

For example, in Oracle Fusion Receivables, the Transactions profile option category groups profile options related to setting how Receivables transactions are to be processed, such as Require Adjustment Reason.

A profile option can be in more than one category.
Profile Option Hierarchies and Levels

Application developers specify at which hierarchy level a profile option is enabled. The predefined profile option hierarchy levels are site, product, and user.

The hierarchy levels specified in the profile option definition determine the context in which a profile option value may be set. If the profile option value at a particular level is updatable, an administrator can update the profile option value for that context.

Note

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

For security, one level in the hierarchy is designated as a user level. A profile option may be enabled at any or all hierarchy levels. When enabled at all levels, the predefined ordering of profile option hierarchy levels gives precedence to the values that are set at the user level over values set at the product and site levels, and precedence to values set at the product level to values set at the site level. If there is no value for the current user, then the product value applies. If there is no value for the user or product, then the site value applies.

The table shows the predefined profile option hierarchy and ordering.

<table>
<thead>
<tr>
<th>Hierarchy Level</th>
<th>Priority When Multiple Levels Set</th>
<th>Effect on Applications</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>Lowest</td>
<td>Affect all applications for a given implementation</td>
<td>Currency for the site is set to Euros.</td>
</tr>
<tr>
<td>Product</td>
<td>Supersedes Site</td>
<td>Affect all applications of a product family such as Financials</td>
<td>Currency for the Financials products set to UK pound sterling.</td>
</tr>
<tr>
<td>User</td>
<td>Highest, supersedes Product</td>
<td>Affect only the experience of the current user</td>
<td>Currency for the user of Financials applications set to US dollars.</td>
</tr>
</tbody>
</table>

You can configure updatable values for profile options at one or more levels depending on which levels are enabled in the profile option definition. When a profile is set at more than one level, higher levels of specificity override lower levels of specificity.

In the example, if the currency setting for the site is UK pounds sterling, but the Financials division works in the Netherlands using the Euro, a manager in the US can override that product level setting at the user level to use US dollars when accessing Financials applications.

In another example, if a profile option called Printer is set only at the site and product levels. When a user logs on, the Printer profile option assumes the value set at the product level, since it is the highest level setting for the profile.
Tip
Set site-level profile option values before specifying values at any other level. The profile option values specified at the site-level work as defaults until profile option values are specified at the other levels.

For more information on the predefined profile options, see assets with the Profile Option type in the Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

Planning Profile Options: Points to Consider

Plan profile options before defining and configuring them.
The following aspects assist you in better planning how to manage profile options.

- Profile option tasks
- Before creating a profile option
- Profile options data model

Profile Option Tasks

Users may be able to set their own profile options, depending on settings in the profile option definition. However, not all profile options are visible to end users, and some profile options, while visible, may not be updated by end users.
The following table lists tasks and considerations relevant to planning profile options.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Role</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning, creating, and editing a new profile option</td>
<td>Applications developer</td>
<td>Since profile options are for permanent settings, do not use profiles options to cache temporary session attributes. Add capacity for user preferences and system configuration. Customize profile options with values, value behaviors, validation, category values, and security. Define the levels at which the profile option is enabled.</td>
</tr>
<tr>
<td>Configure values in an existing profile option</td>
<td>Applications developer, application administrator, and implementation consultant</td>
<td>Manage the values for existing profile options.</td>
</tr>
<tr>
<td>Create and edit profile option categories</td>
<td>Applications developer, application administrator, and implementation consultant</td>
<td>Manage categories for organizing existing profile options.</td>
</tr>
</tbody>
</table>

Note
Since a profile option enables a behavior in an application user interface or across applications, a value change made by an end user is reflected in the UI page for managing profile option values.

### Before Creating a Profile Option

Profile options are best defined for managing configuration data centrally and influencing the behavior of applications.

If the purpose of a profile option setting is specific to a piece of data (typically setup data), it is best implemented as an attribute of that data.

Do not use profile options for behavior that is not configurable.

Profile options exist independent of role.

Do not use profile options to implement function security. For example, an application should not check for a profile option value set to yes to provide access to a page. Do not use profile options to implement data security, such as a profile option value that must be set to a specific value to provide view access to an entity.

Do not use profile options to capture a dynamic system states, such as data stored in a temporary table. Use Global Variables for temporary states instead.

Evaluate if there is a genuine need before creating a profile option. Do not force users to make a decision about an aspect of their application use that is of no concern.

Evaluating need includes looking for duplicate or similar profile options, even in other products, before creating a new one. For example, you do not need multiple profile options to choose a preferred currency.

### Profile Options Data Model

The profile option data model illustrates the relationships among profile option elements.

The figure shows the data model of profile option entities.
For more information about planning profile options, see the Oracle Fusion Applications Developer’s Guide.

Managing Profile Options: Points to Consider

A profile option definition consists of a name for the profile option and valid values. It is defined within a module of the application taxonomy. Application developers manage profile options to create new profile options or modify existing profile option definitions, which includes specifying the levels at which a profile option is enabled and defining values. Implementation consultants and application administrators configure existing profile options by managing the profile option’s updatable values, and creating categories that group profile options.

Configuring a Profile Option

A profile option definition includes information about the owning application and module in the application taxonomy. A start or end date, or both may limit when a profile option is active. The profile option definition may include an SQL validation statement that determines which values are valid, and the hierarchy levels at which the profile option is enabled and updatable.

To be visible to users, a profile option must be user enabled. You can also allow user updates of the profile option, which means users can make changes to the validation and the profile option level information.

Profile option levels specify at which context level profile values may be enabled or updated.

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

SQL Validation

The SQL validation of the profile option definition determines what valid profile option values are available. In the absence of validation, any value is valid.

For example, SQL validation provides a means of defining a list of values for the valid values of the profile option. The SQL validation can use lookups to provide the valid values for profile options, such as the lookup codes of the YES_NO lookup type.

With a profile option called DEFAULT_LANGUAGE, you can configure the following validation.

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.

Exporting and Importing Customizations: Overview

Read this chapter to learn about how to export and import customizations across Oracle application instances on the same release.
In this chapter, you will learn about:

- How to use customization migration to move customizations
- How to create and apply customizations using customization set

**Note**

You can also import and export custom object transactional data as opposed to metadata. Two processes exist to enable the importing and exporting of object data: file-based import and bulk export. After you create custom objects in Application Composer, you must then manually generate the object artifacts required for both file-based import and bulk export. This process, which is unrelated to the exporting and importing of customizations, is described in a separate topic, "Importing and Exporting Custom Objects: Explained."

To access customization set migration (CSM) in the source or target environment, select **Navigator - Tools - Customization Migration.** CSM is accessible as a link called **Customization Migration** under the **Tools** menu item.

### 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 runtime page. Flexfields in highlight mode display an Information icon button to access details about the flexfield, an Edit icon button to manage the flexfield, and an Add Segment icon button to add flexfield segments.

All segments of a single flexfield are grouped together by default. The layout and positions of the flexfield segments depend on where the application developer places the flexfield on the page. Flexfields may also be presented in a separate section of the page, in a table, or on their own page or subwindow.

You can use Oracle Composer to edit the layout, position, or other display features of the flexfield segments.

Configuring Flexfields: Overview

Configuring a flexfield ranges from identifying the need for extending a business object with custom attributes to integrating the custom attributes into the deployment. In the case of key flexfields, configuring the flexfield involves identifying value set assignments and determining segment structures.

Overall Process for Configuring Custom Attributes

For descriptive and extensible flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan flexfield validation.
4. Define the attributes by configuring the flexfield segments.
   a. Use the Manage Extensible Flexfields or Manage Descriptive Flexfields tasks, or use the Configure icon button directly on the page where the flexfield is highlighted. For simple configurations, use the Add Segment, Add Context Value, and Edit Segment icon buttons directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.
5. Deploy the flexfield to the mainline to display the custom attributes on the application pages and to make them available for integration with other tools such as Oracle Business Intelligence.
6. Perform the necessary steps to integrate the custom attributes into the technology stack.

A simple configuration is limited to such actions as adding a format-only field or adding a field with a basic list of values.
Overall Process for Configuring Custom Keys

Using key flexfields, you can configure intelligent key codes comprised of meaningful parts according to your business practices. You configure the key flexfield to have one segment for each part that makes up your key code.

For key flexfields, the overall configuration process involves the following:

1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
2. Plan the flexfield configuration.
3. Plan the flexfield validation.
4. Define the value sets before configuring the key flexfield segments by going to the Manage Value Sets task.
5. Define the key flexfield structures and their segments, and define structure instances for each structure.
   a. Use the Manage Key Flexfields task or the **Configure** icon button directly on the page where the flexfield is highlighted.
   b. Optionally, validate the flexfield configuration.
   c. Optionally, deploy the flexfield to a sandbox for initial testing.
6. Deploy the flexfield to the mainline to display it on the application pages and to make it available for integration with other tools such as Oracle Business Intelligence.
7. Perform the necessary steps to integrate the flexfield into the technology stack.

Flexfields at Run Time: Explained

Many business objects in Oracle Fusion applications have an associated descriptive or extensible flexfield with which you can create custom attributes for the business object. Some business objects have an associated key flexfield for configuring flexible multiple part keys.

The following aspects are important in understanding flexfields at run time:

- Finding flexfields on a page
- Why no flexfields are on a page

Finding Flexfields on a Page

At run time, the custom attributes you define as extensible and descriptive flexfield segments appear in the application page just like any other attribute. Key flexfields typically appear in the application page as a field with a key flexfield icon, where the field’s value is actually a collection of segments. In some pages, one or more key flexfield segments may be displayed in the page like any other attribute. Thus, when viewing the page in standard mode, in many cases
you may not be able to discern which fields are flexfield segments, or whether flexfields are available to configure on the page.

Use the Highlight Flexfields feature to render the page in a special mode that lets you view:

- Where, if any, flexfields are available on your page
- Which, if any, of the fields on your page are flexfield segments rather than out-of-the-box fields

To obtain information about the flexfields on a page, open the page and choose Highlight Flexfields from the Administration menu. Hover over the Information icon button next to the highlighted fields to display information about the flexfield. Choose Unhighlight Flexfields from the Administration menu when you no longer want to see the highlighted flexfields.

When you click the Configure Flexfield icon button for a highlighted flexfield, the applicable Manage Flexfields task is displayed for that flexfield. For simple configurations, you can click the Add Context Value icon button to add a context value, or click the Add Segment or Edit Segment icon buttons to add or edit a global segment or a context-sensitive segment that doesn't require advanced configuration.

**Note**

Not all flexfields are available for creating custom attributes. Consult the product-specific documentation in Oracle Fusion Applications Help to verify whether there are any restrictions on using the flexfield.

**Why No Flexfields Are on a Page**

For a flexfield to be available in the page, it must be registered by developers. If a flexfield is available, you may configure segments. The segments appear on the page only after you have successfully deployed the flexfield. For information about registering flexfields, see the Oracle Fusion Applications Developer’s Guide. Some business objects haven’t been designed to support flexfields. For information about how to enable business objects with flexfield capability, see Getting Started with Flexfields in the Oracle Fusion Applications Developer’s Guide.

**Note**

Oracle Sales Cloud doesn’t support flexfields.

To add custom attributes to these applications, use Application Composer. For more information, see the "Editing an Object: Explained" section in Oracle Sales Cloud: Extending Sales.

**Customizing Flexfields Using Page Composer: Explained**

Using Page Composer, you can create customizations to flexfields that are specific to a page.
In Page Composer, to customize:

- Extensible flexfields, open the page in Source view, and look for a region that is bound to an EffContextsPageContainer task flow. This is the container for the extensible flexfield attributes and contexts. To view the flexfield code and identifying information, open the properties panel for the region. To customize any component within the region, select the desired tag and click Edit.

- Descriptive flexfields, open the page in Source view, and look for `<descriptiveFlexfield>` elements. Open the properties panel for the element to view the flexfield code and identifying information. Within the properties panel, you may customize properties for the global and context-sensitive segments or re-order the segments on the page.

**Accessing Flexfield Management Tasks: Procedures**

You can configure and manage flexfields by highlighting them on an application page and using the available on-screen configuration tools. Alternatively, you can access product-specific flexfield tasks or global flexfield management tasks.

**Accessing Flexfield Management Tasks through the Run time Page**

You can identify flexfields on the run time application page where they are implemented.

1. Navigate to an application page.
2. Choose Highlight Flexfields from the Administration menu in the global area of Oracle Fusion Applications.
3. View the available flexfields highlighted on the page. If any of the fields on the page are custom fields configured as part of a flexfield, they also appear highlighted.
4. To edit a flexfield, use the:
   - Configure Flexfield icon button to access the flexfield management task pages for extensive configuration to the flexfield and its segments.
   - Add Segment icon button to access the subwindow for adding segments with limited configuration to descriptive flexfields.
   - Edit Segment icon button to access the subwindow for limited configuration changes to descriptive flexfield segments.

**Accessing Flexfield Management Tasks through Setup and Maintenance**

Manage flexfields using tasks you access by starting in the Setup and Maintenance Overview page which is available from the Navigator or the Administration menu.

To access tasks for configuring flexfields and value sets, you must be provisioned with roles that entitle you to access the Define Flexfields task list or tasks for managing product-specific flexfields. Contact your security administrator for
details. For information about product-specific flexfield tasks, such as Manage Purchasing Descriptive Flexfields, consult the product-specific documentation in Oracle Fusion Applications Help.

To access the flexfield management tasks and search for existing flexfields, perform the following steps:

1. Choose Setup and Maintenance from the Administration menu in the global area of Oracle Fusion Applications.
2. Search for Define Flexfields in the All Tasks tab.

Tip

- Use the Business Object parameter to search:
  - Application Key Flexfields, Application Descriptive Flexfields, and Application Extensible Flexfields to find all tasks related to flexfields.
  - Application Flexfield Value Set to find all tasks related to value sets.
- To manage any:
  - Flexfield across all Oracle Fusion Applications products, search for the Define Flexfields task list and access the Manage Descriptive Flexfields, Manage Extensible Flexfields, and Manage Key Flexfields tasks.
  - Value set across all Oracle Fusion Applications products, search for the Define Flexfields task list and access the Manage Value Sets task.

Restriction

If you are configuring key flexfields, search for and access the Manage Value Sets task to set up value sets before accessing the Manage Key Flexfields task.

3. Expand the task list to view the included tasks.
4. Click the Task icon button to open the manage flexfield pages.
5. Search for all or specific flexfields.
6. In the search results, select the flexfield.
7. Use the Edit action to open pages for viewing and configuring the flexfield. Access to managing value sets is available within the tasks for managing descriptive and extensible flexfields.

Note

Access to managing value sets is:

- Available within the tasks for managing descriptive and extensible flexfields.
- Not available within the tasks for managing key flexfields. Therefore, configure value sets prior to configuring your key flexfield.
Administrators configure flexfield segments to capture data that represents the values of attributes. Flexfield segments represent attributes of entities (business objects). Most business objects are enabled for descriptive flexfields. Some business objects are enabled for extensible flexfields.

For example, an airline manufacturer might require very specific attributes for their orders that aren’t provided by the out-of-the-box implementation of an order. Because a flexfield exists for the order business object, you can use it to create and configure the desired attribute.

The figure shows the layers of a flexfield: the business entity table and metadata in the database, business components that are Application Development Framework (ADF) objects or ADF business component (ADFbc) objects derived from the metadata and stored in the Metadata Services Repository (MDS), and the user interface where the input fields defined by the flexfield segments are rendered. The flexfield definition consists of all the metadata defined during configuration and stored in the database.

Application developers create a flexfield and register it so that it is available for configuration. Administrators and implementation consultants configure segments and other properties of the available flexfields. This information is
stored as additional flexfield metadata in the database. Deploying the flexfield generates ADF business components based on the flexfield metadata in the database.

The following aspects are important in understanding how flexfields and Oracle Fusion Applications architecture work together:

- Integration
- Deployment
- Import and Export
- 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.

**Flexfields and Value Sets: Highlights**

Before you use flexfields to create custom attributes, you should be familiar with the Oracle Fusion application architecture that enables customization, customization layers, and the customization lifecycle.

In addition to the extensive information in the Oracle Fusion Applications Help about configuring flexfields that are already available for configuration, consider the resources below for adding flexfields to business components and alternatives to flexfields where flexfields cannot be enabled.

To assess the flexfields available in a deployment of Oracle Fusion Applications, see assets of type: flexfield in the Oracle Enterprise Repository at http://fusionappsoer.oracle.com.


**Restriction**

Don’t use Oracle JDeveloper to customize flexfields.

**Before Configuring Flexfields**

You can add custom attributes to a business object using a flexfield, if a flexfield has been registered for that object by developers.

- For information about registering flexfields to business objects, refer to the Oracle Fusion Applications Developer’s Guide.

  See: Getting Started with Flexfields

- The user interface page for a business object that a developer extends to support a flexfield must be enabled to display the custom attributes defined by the flexfield.

  See: Adding Descriptive Flexfield UI Components to a Page

  See: Employing an Extensible Flexfield on a User Interface Page

- For Oracle Sales Cloud, use Application Composer to add custom attributes instead of using descriptive and extensible flexfields. For more information, refer to Oracle Sales Cloud: Extending Sales.

  See: Application Composer: Introduction

- For information about displaying translated values of a table-validated value set from the value column for the run time session’s locale, refer to the Oracle Fusion Applications Developer’s Guide.
See: Using Multi-Language Support Features

Security

• For an understanding of data security when considering the consequences of applying data security to value sets, refer to the Oracle Fusion Applications Security Guide.

See: Data Security

Deploying Flexfields

• To examine the artifacts of a deployed flexfield configuration that you exported using the exportMetadata WLST command, refer to the Oracle Fusion Applications Extensibility Guide.

See: Exporting Customizations

• For information about synchronizing the updated XML schema definition (XSD) files in the metadata services (MDS) repositories for each service-oriented architecture (SOA) application, refer to the Oracle Fusion Applications Extensibility Guide.

See: Customizing SOA Composite Applications

• For information about incorporating a deployed flexfield into the technology stack, such as customizing the pages, integrating with Oracle Business Intelligence, or integrating into Web Services and service-oriented architecture SOA infrastructure, refer to the Oracle Fusion Applications Concepts Guide.

See: Oracle Fusion Middleware Components

• Oracle ADF services used by SOA composites expose the Web Services Description Language (WSDL) schemas where deployed flexfields are stored.

See: Oracle Fusion Middleware Developer’s Guide for Oracle SOA Suite

Oracle Business Intelligence

• For more information about importing and propagating your flexfield changes, refer to the Oracle Fusion Applications Extensibility Guide.

See: Customizing the Oracle BI Repository (RPD)

• For information about importing business intelligence-enabled flexfield changes into the Oracle Business Intelligence repository, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

Flexfield Management

Managing Flexfields: Points to Consider

Managing flexfields involves registering, planning, and configuring flexfields.
You plan and configure the registered flexfields provided in your applications by applications developers. How you configure flexfield segments determines how the flexfield segments appear to end users. Optionally, you can customize the UI page to change how the flexfield segments appear to end users on that page.

The figure shows the processes involved in making flexfields available to end users. The tasks in the Define Flexfields activity let administrators configure and deploy flexfields. If you deploy a flexfield to a sandbox and decide to apply the configuration to the mainline, select the flexfield in the Manage Flexfields tasks of the Define Flexfields activity and deploy the flexfield in the mainline so that it is available to users.

Consider the following aspects of managing flexfields:

- Registering flexfields
- Planning flexfields
- Configuring flexfields
- Enabling a flexfields segment for business intelligence
- Deploying flexfields
- Optionally changing a flexfield segment’s appearance in a user interface page
- Identifying flexfields on a 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 runtime page.
- Use the **Configure Flexfield** icon button to manage a flexfield, such as change a segment’s sequence number, or configure a flexfield segment’s business intelligence label.
- Use the **Add Segment** icon button to add descriptive flexfield segments and context values, or extensible flexfield segments.
Configuring a flexfield includes the following:

- Defining value sets against which the values entered by end users are validated
- Defining the structure or context of the segments in the flexfield
- Specifying the identifying information for each segment
- Specifying the display properties such as prompt, length and data type of each flexfield segment
- Specifying valid values for each segment, and the meaning of each value within the application

**Tip**

You can create value sets while creating descriptive and extensible flexfield segments. However, define value sets before configuring key flexfield segments that use them, because you assign existing value sets while configuring key flexfield segments.

When creating table-validated, independent, dependent, or subset value sets while creating descriptive and extensible flexfield segments, you can optionally specify to display the description of the selected value to the right of the segment at 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 descriptive, extensible, and key 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:

- 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>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</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>
<tr>
<td>Instruction help text</td>
<td>The field-level instruction help text to display for the field.</td>
</tr>
<tr>
<td></td>
<td>Use instruction help text to provide directions on using the field. If instruction help text is specified, it is displayed in an in-field help note window that appears when users give focus to or hover over the field.</td>
</tr>
</tbody>
</table>

Properties Related to Search

Extensible flexfield segments can be marked as selectively required in search using the indexed property. The indexed property requires end users to enter a value before conducting a search on the attribute represented by the indexed segment. A database administrator must create an index on the segment column representing the indexed attribute.

Range Validation of Segments

Range validation enables you to enforce an arithmetic inequality between two segments of a flexfield. For example, a product must be ordered before it can be shipped. Therefore, the order date must be on or before the ship date, and consequently the order date segment value must be less than or equal to the ship date segment value. You can use range validation to ensure this relationship.

The conditions for range validation are as follows:

- Segments must be configured for range validation in pairs, one with the low value and one with the high value.
- Both segments must be of the same data type.
- Both segments must be parts of the same structure in a key flexfield or parts of the same context in a descriptive flexfield or extensible flexfield.
- The low value segment must have a lower sequence number than the high value segment.
- Non-range validated segments can exist between a range validated pair, but range validated pairs cannot overlap or be nested.

You can configure as many range validated pairs as you want within the same flexfield. Your application automatically detects and applies range validation to the segment pairs that you define, in sequence order. It must encounter a low
value segment first, and the next range validated segment that it encounters must be a high value segment. These two segments are assumed to be a matching pair. The low value and the high value can be equal.

**Rule Validation of Segment Values**

Validation rules on descriptive and extensible flexfield segments determine how an attribute is validated. The value entered for an attribute on a business object may need to match a specified format or be restricted to a list of values. Use a value set to specify the validation rules.

Value set validation is required for global segments and context-sensitive segments, and optional for context segments. In the case of context segments, the application may validate an input value instead of the value set validating the input value against the context segment. However, the application input values must match exactly the valid context segment values. If the context segment values are a superset or subset of the input values, you must assign a table validated value set or independent value set to validate context values.

When you configure a descriptive flexfield segment, you can specify a constant to use for setting the initial value. The initial value can be an available parameter. For every planned segment, list the constant value or parameter, if any, to use for the initial value.

**Naming Conventions**

Enter a unique code, name, and description for the segment. These properties are for internal use and not displayed to end users. You can’t change the code after the segment is created.

The Application Programming Interface (API) name is a name for the segment that isn’t exposed to end users. The API name is used to identify the segment in various integration points including web services, rules, and business intelligence. Use alphanumeric characters only with a leading character. For example, enter a code consisting of the characters A-Z, a-z, 0-9 with a non-numeric leading character. The use of spaces, underscores, multi-byte characters, and leading numeric characters isn’t permitted. You can’t change the API name after the segment has been created.

**Flexfields Segments: How They Are Rendered**

Flexfield segments appear on pages as attributes of business objects.

**Settings That Affect Flexfield Segment Display**

When you configure flexfield segments, the value you enter for the segment’s display type determines how the segment appears on the run time page.

**How Display Type Values Appear**

The figure shows how display types appear at run time.

In the following figure, identify the display type by letter when referring to the table of descriptions for check box, drop-down list, list of values, pop-up list of values, and radio button group.
In the following figure, identify the display type by letter when referring to the table of descriptions for radio button group, text area, text box, and date/time.

A. Check Box

B. Drop-down List

C. List of Values

D. Pop-up List of Values

E. Radio Button Group

F. Text Area

G. Text Box

H. Date/Time

The table describes each display type. The Example column refers to the figures above.
<table>
<thead>
<tr>
<th>Display Type</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Box</td>
<td>A</td>
<td>The field is displayed as a check box. If the end user selects the checkbox, the checked value is used. Otherwise, the unchecked value is used.</td>
</tr>
<tr>
<td>Drop-down List</td>
<td>B</td>
<td>The field displays a dropdown list of values from which the end user can select a value.</td>
</tr>
<tr>
<td>List of Values</td>
<td>C</td>
<td>The field displays a dropdown list of values from which the end user can select a value. The user can also click Search to find more values.</td>
</tr>
<tr>
<td>Pop-up List of Values</td>
<td>D</td>
<td>The field displays as a text field with a Search icon button. The end users can type a value in the text field or they can click the Search icon button to open a subwindow for searching.</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>E</td>
<td>The field is displayed as a set of radio buttons. The end user can select one button. Selecting a button deselects any previously selected button in the set.</td>
</tr>
<tr>
<td>Text Area</td>
<td>F</td>
<td>The field is displayed as a text area in which the end user can type multiple lines of text. The display width and height specify the visible width and number of lines in the text area, respectively.</td>
</tr>
<tr>
<td>Text Box</td>
<td>G</td>
<td>The field is displayed as a text field in which the end user can type a single line of text. The display width controls the width of the text box.</td>
</tr>
<tr>
<td>Date/Time</td>
<td>H</td>
<td>The field enables the end user to enter a date if the data type is Date, or a date and time if the data type is Date Time. The user can select the date from a calendar. If the data type is Date Time, the field also displays fields for specifying the hour, minutes, seconds, AM or PM, and time zone.</td>
</tr>
</tbody>
</table>
### Display Type

<table>
<thead>
<tr>
<th>Display Type</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hidden</td>
<td></td>
<td>The field isn’t displayed.</td>
</tr>
</tbody>
</table>

### Flexfields and Value Sets: How They Work Together

Value sets are specific to your enterprise. When gathering information using flexfields, your enterprise’s value sets validate the values that your users enter based on how you defined the value set.

You can assign a value set to any number of flexfield segments in the same or different flexfields. Value set usage information indicates which flexfields use the value set.

The following aspects are important in understanding how flexfields and value sets work together:

- Defining value sets
- Shared value sets
- Deployment

### Defining Value Sets

As a key flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfields, you can define value sets when adding or editing a segment.

**Caution**

Be sure that changes to a shared value set are compatible with all flexfield segments that use the value set.

### Shared Value Sets

When you change a value in a shared value set, the change affects the value set for all flexfields that use that value set. The advantage of a shared value set is that a single change propagates to all usages. The drawback is that the change shared across usages may not be appropriate in every case.

### Value Set Values

To configure custom attributes to be captured on the value set values screen in the Manage Value Sets task, configure the Value Set Values descriptive flexfield. The object’s code is FND_VS_VALUES_B. This flexfield expects the context code to correspond to the value set code. For each value set, you can define a context whose code is the value set code, and whose context-sensitive segments will be shown for the values of that value set. By default the context segment is hidden since it defaults to the value set code and is not expected to be changed.

You can also define global segments that will be shown for all value sets. However, this would be quite unusual since it would mean that you want to capture that attribute for all values for all value sets.
Deployment

When you deploy a flexfield, the value sets assigned to the segments of the flexfield provide end users with the valid values for the attributes represented by the segments.

Defaulting and Deriving Segment Values: Explained

To populate a flexfield segment with a default value when a row is created, specify a default type of constant or parameter and a default value.

To synchronize a segment’s value with another field’s value whenever it changes, specify the derivation value to be the flexfield parameter from which to derive the attribute’s value. Whenever the parameter value changes, the attribute’s value is changed to match. If you derive an attribute from a parameter, consider making the attribute read-only, as values entered by users are lost whenever the parameter value changes.

When defaulting or deriving a default value from a parameter, only those attributes designated by development as parameters are available to be chosen.

Different combinations of making the segments read only or editable in combination with the default or derivation value or both, have different effects.

Initial 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>Constant</td>
<td>Yes</td>
<td>Yes</td>
<td>Default segment value</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>No</td>
<td>The default segment value is the parameter’s default value</td>
<td>N/A</td>
</tr>
</tbody>
</table>
### FAQs for Flexfield Management

**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.

**How can I enable flexfield segments for Oracle Social Network Cloud Service?**

Descriptive flexfield segments can be enabled for integration with Oracle Social Network Cloud Service. When you manage Oracle Social Network Objects during setup and maintenance, search for the business object that includes descriptive flexfields, and select the business object attributes that are defined as flexfield segments.

### Flexfield Deployment

**Flexfield 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: 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 redeploy 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>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.

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.

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><strong>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.</strong></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><strong>Deploy a single flexfield regardless of deployment status</strong></td>
<td>Deploy a single flexfield regardless of deployment status.</td>
</tr>
<tr>
<td><strong>Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.</strong></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><strong>Displays MDS label of flexfield changes for viewing and deleting patching labels.</strong></td>
<td>Displays MDS label of flexfield changes for viewing and deleting patching labels.</td>
</tr>
<tr>
<td><strong>Displays list containing flexfields that aren’t deployed or failed deployment.</strong></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:**

```
wlst.cmd
```

Connect to the server, replacing the user name and password arguments with your WebLogic Server user name and password.

```
connect('wls_username', 'wls_password', 'wls_uri')
```

The values must be wrapped in single-quotes. The `wls_uri` value is typically `T3://localhost:7101`.

For more information on the WLST scripting tool, see the Oracle Fusion Middleware Oracle WebLogic Scripting Tool.
Using the deployFlexForApp Command

The **deployFlexForApp** command translates the product application’s predefined flexfield metadata into artifacts in the MDS repository.

**Important**

This command is run automatically when you provision applications. However, after custom applications development, you must run the **deployFlexForApp** command after you configure your application to read the flexfield artifacts from the MDS repository and before you log into the application for the first time, even if there is no predefined flexfield metadata.

This command 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.

```python
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.

```python
deployPatchedFlex()
```

Execute the following command to deploy all flexfields that have either a READY status or an ERROR status.

```python
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:

```python
deleteFlexPatchingLabels(infoOnly='true')
```

**Using the validateFlexDeploymentStatus Command**

The `validateFlexDeploymentStatus()` WLST command checks the deployment status of all flexfields in an Oracle Fusion Applications deployment.

```python
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.

```java
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.

The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.

For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

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

**Defining Value Sets: Critical Choices**

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

As a flexfield guideline, define value sets before configuring the flexfield, because you can assign value sets to each segment as you configure a flexfield. With descriptive and extensible flexfield segments, you can create value sets when adding or editing a segment on the 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 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.

**Planning Value Sets: Points to Consider**

The value sets you create and configure depend on the valid values on the business object attributes that will use the value set. When creating value sets, you first give the value set a name and description, and then define the valid values of the set.

The following aspects are important in planning value sets:

- List of values
- Plain text input
- Value ranges
- Value format specification
- Security

**List of Values**

You can use one of the following types of lists to specify the valid values for a segment:

- Table column
- Custom list
  - Subset of custom list
  - Dependent custom list

If the valid values exist in a table column, use a table value set to specify the list of values. To limit the valid values to a subset of the values in the table, use a SQL WHERE clause. Table value sets also provide some advanced features, such as enabling validation depending on other segments in the same structure.

Use an independent value set to specify a custom set of valid values. For example, you can use an independent value set of Mon, Tue, Wed, and so forth to validate the day of the week. You can also specify a subset of an existing independent value set as the valid values for a segment. For example, if you
have an independent value set for the days of the week, then a weekend subset can be composed of entries for Saturday and Sunday.

Use a dependent value set when the available values in the list and the meaning of a given value depend on which independent value was selected for a previously selected segment value. For example, the valid holidays depend on which country you are in. A dependent value set is a collection of value subsets, with one subset for each value in a corresponding independent value set.

For lists of values type value sets, you can additionally limit the valid values that an end user can select or enter by specifying format, minimum value, and maximum value. For list of values type value sets, you can optionally implement value set data security. If the Oracle Fusion applications are running in different locales, you might need to provide different translations for the values and descriptions.

Plain Text Input

Use a format-only value set when you want to allow end users to enter any value, as long as that value conforms to formatting rules. For example, if you specify a maximum length of 3 and numeric-only, then end users can enter 456, but not 4567 or 45A. You can also specify the minimum and maximum values, whether to right-justify, and whether to zero-fill. With a format-only value set, no other types of validation are applied.

Value Ranges

You can use either a format-only, independent, or dependent value set to specify a range of values. For example, you might create a format-only value set with Number as the format type where the end user can enter only the values between 0 and 100. Or, you might create a format-only value set with Date as the format type where the end user can enter only dates for a specific year, such as a range of 01-JAN-93 to 31-DEC-93. Because the minimum and maximum values enforce these limits, you need not define a value set that contains each of these individual numbers or dates.

Value Format

Flexfield segments commonly require some kind of format specification, regardless of validation type. Before creating a value set, consider how you will specify the required format.

The following table shows options for validation type and value data type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value data type</td>
<td>Character, Number, Date, Date Time.</td>
</tr>
<tr>
<td>Value subtype</td>
<td>Text, Translated text, Numeric digits only, Time (20:08), Time (20:08:08).</td>
</tr>
<tr>
<td></td>
<td>An additional data type specification for the Character data type for the Dependent, Independent, and Format validation types.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Maximum length</td>
<td>Maximum number of characters or digits for Character data type.</td>
</tr>
<tr>
<td>Precision</td>
<td>Maximum number of digits the user can enter.</td>
</tr>
<tr>
<td>Scale</td>
<td>Maximum number of digits that can follow the decimal point.</td>
</tr>
<tr>
<td>Uppercase only</td>
<td>Lowercase characters automatically changed to uppercase.</td>
</tr>
<tr>
<td>Zero fill</td>
<td>Automatic right-justification and zero-filling of entered numbers (affects values that include only the digits 0-9).</td>
</tr>
</tbody>
</table>

**Caution**

You cannot change the text value data type to a translated text value subtype after creating a value set. If there is any chance you may need to translate displayed values into other languages, choose Translated text. Selecting the Translated text subtype doesn't require you to provide translated values.

**Value Sets for Context Segments**

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context's column length. If you use a table value set, the value set cannot reference flexfield segments in the value set's WHERE clause other than the flexfield segment to which the value set is assigned.

**Security**

When enabling security on a value set, the data security resource name is an existing value set or one that you want to create. The name typically matches the code value for the value set.

**Restriction**

You cannot edit the data security resource name after you save your changes.

**Table-Validated Value Sets and Bind Variables: Points to Consider**

After you assign a value set to a flexfield, you can use bind variables in the WHERE clause.

The following bind variables refer to flexfield elements:
• :{SEGMENT.<segment_code>}
• :{CONTEXT.<context_code>;SEGMENT.<segment_code>}
• :{VALUESET.<value_set_code>}
• :{FLEXFIELD.<internal_code>}
• :{PARAMETER.<parameter_code>}

**Segment Code**

:`{SEGMENT.<segment_code>}`

This bind variable refers to the ID or value of a segment where `<segment_code>` identifies the segment. Where referring to the ID, the value set is ID-validated. Where referring to the value, the value set isn’t ID-validated. The data type of the bind value is the same as the data type of the segment’s column.

For both descriptive and extensible flexfields, the segment must be in the same context as the source segment. The source segment contains the WHERE clause. For descriptive flexfields, if the segment is global, then the source segment must be global.

The segment must have a sequence number that is less than the sequence number of the target segment with this bind variable. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment. For example, the values to select from a CITIES table might depend upon the selected country. If SEGMENT1 contains the country value, then the WHERE clause for the CITIES table might be `<country_code> = :{SEGMENT.SEGMENT1}`.

**Context Code**

:`{CONTEXT.<context_code>;SEGMENT.<segment_code>}`

This bind variable, which is valid only for extensible flexfields, refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of a segment that is in a different context than the target segment (the segment with the WHERE clause).

- The `<context_code>` identifies the context and must be in the same category or in an ancestor category. It cannot be a multiple-row context.
- The `<segment_code>` identifies the segment. The data type of the bind value is the same as the data type of the segment’s column.

**Tip**

The target segment should appear in the UI after the source segment to ensure the source segment has a value. If the target segment’s context is a single-row context, the source and target segments must be on separate pages and the target page must follow the source page.

This bind variable is useful when the set of valid values depends on the value of a segment in another context. For example, the values to select from a
CERTIFICATION table for a segment in the Compliance and Certification context might depend on the value of the country segment in the Manufacturing context.

**Value Set Code**

`:VALUESET.<value_set_code>`

This bind variable refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of the segment that is assigned to the value set that is identified by the `value_set_code`. The data type of the bind value is the same as the data type of the segment’s column.

The segment must have a sequence number that is less than the sequence number of the segment with this bind variable. If more than one segment is assigned to the value set, the closest prior matching segment will be used to resolve the bind expression. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment and that segment code can vary, such as when the value set is used for more than one context or flexfield. For example, the values to select from a CITIES table might depend upon the selected country. If the value set for the segment that contains the country value is COUNTRIES, then the WHERE clause for the CITIES table might be `<county_code> = :VALUESET.COUNTRIES`.

**Flexfield Internal Code**

`:FLEXFIELD.<internal_code>`

This bind variable refers to an internal code of the flexfield in which the value set is used, or to a validation date. The `internal_code` must be one of the following:

- **APPLICATION_ID** - the application ID of the flexfield in which this value set is used. The data type of APPLICATION_ID and its resulting bind value is NUMBER.

- **DESCRIPTIVE_FLEXFIELD_CODE** - the identifying code of the flexfield in which this value set is used. The data type of DESCRIPTIVE_FLEXFIELD_CODE and its resulting bind value is VARCHAR2. Note that you use this string for both descriptive and extensible flexfields.

- **CONTEXT_CODE** - the context code of the flexfield context in which this value set is used. The data type of CONTEXT_CODE and its resulting bind value is VARCHAR2.

- **SEGMENT_CODE** - the identifying code of the flexfield segment in which this value set is used. The data type of SEGMENT_CODE and its resulting bind value is VARCHAR2.

- **VALIDATION_DATE** - the current database date. The data type of VALIDATION_DATE and its resulting bind value is DATE.

**Flexfield Parameters**

`:PARAMETER.<parameter_code>`
This bind variable refers to the value of a flexfield parameter where parameter_code identifies the parameter. The data type of the resulting bind value is the same as the parameter’s data type.

**Note**

You cannot assign a table value set to a context segment if the WHERE clause uses VALUESET.value_set_code or SEGMENT.segment_code bind variables.

### Table-Validated Value Set: Worked Example

In an application user interface, you want to display a list of values that allow customers to enter satisfaction scores. The value column name is 1, 2, 3, 4, 5 and the value column description is Extremely Satisfied, Satisfied, and so on. Users can pick the appropriate value or description which stores the corresponding name so the name value can be used in a calculation expression.

In this case, you can use the FND_LOOKUPS table as the basis for a table-validated value set. The lookup meaning corresponds to the Value Column Name and the lookup description corresponds to the Description Column Name. The properties of the value set are as follows:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM clause</td>
<td>FND_LOOKUPS</td>
</tr>
<tr>
<td>WHERE clause</td>
<td>lookup_type = ‘CN_XX_CUST_SATISFACT SCORE’</td>
</tr>
<tr>
<td>ID column</td>
<td>lookup_code</td>
</tr>
<tr>
<td>Value column</td>
<td>meaning</td>
</tr>
<tr>
<td>Description column</td>
<td>description</td>
</tr>
<tr>
<td>Enable Flag column</td>
<td>enabled_flag</td>
</tr>
<tr>
<td>Start Date column</td>
<td>start_date_active</td>
</tr>
<tr>
<td>End Date column</td>
<td>end_date_active</td>
</tr>
<tr>
<td>Order by</td>
<td>display_sequence</td>
</tr>
</tbody>
</table>

After completing this task, you should have created your customer satisfaction value set for the Incentive Compensation page of your implementation project.

**Creating a Value Set Based on a Lookup**

1. From the Setup and Maintenance work area, find the Manage Value Sets task and click the Go to Task icon button.
2. On the Manage Value Sets page, click the Create icon button.
3. On the Create Value Set page, enter the following values:
a. In the Value Set Code field, enter  
   CN_XX_CUSTOMER_SATISFACTION_SCORES  
b. In the Description field, enter Customer satisfaction score.  
c. In the Module field, select Search....  
d. In the Search and Select: Module subwindow, enter Incent in the User Module Name field  
e. Select Incentive Compensation.  
f. Click OK.  

4. On the Create Value Set page, enter the following values:  
a. In the Validation Type field, select Table.  
b. In the Value Data Type field, select Character.  
c. In the Definition section FROM Clause field, enter FND_LOOKUPS.  
d. In the Value Column Name field, enter DESCRIPTION.  
e. In the Description Column Name field, enter MEANING.  
f. In the ID Column Name field, enter LOOKUP_CODE.  
g. In the Enabled Flag Column Name field, enter 'Y'.  
h. In the Start Date Column Name field, enter START_DATE_ACTIVE.  
i. In the End Date Column Name field, enter END_DATE_ACTIVE.  
j. In the WHERE Clause field, enter LOOKUP_TYPE = 'CN_XX_CUST_SATISFACT_SCORE'.  

5. Click Save and Close.  
6. In the Manage Value Sets page, click Done.  

**Adding Attributes to the Manage Value Sets Page: Procedures**  

For independent, dependent, and subset value sets, you can add attributes to a value set. The attributes appear in the Manage Value Sets UI for capturing additional information about each valid value, such as its purpose.  

Typically, these attributes are used to capture internal information. To display attributes on an application page, you must programmatically modify the application to access them.  

1. Find the FND_VS_VALUES_B flexfield using the Manage Descriptive Flexfields task.  
2. Open FND_VS_VALUES_B for editing.  
3. Click Manage Contexts.  
4. Create a new context and use the value set code for the context code.  
5. Add the new attributes as context-sensitive segments.  
6. Deploy FND_VS_VALUES_B to the run time.  
7. Sign out and sign back in.  
8. Open the Manage Value Sets page to view the new attributes.
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.

**Planning Descriptive Flexfields: Points to Consider**

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users. Review the tools and tasks available for

---

14-88 Oracle Sales Cloud Implementing Customer Data Management
managing flexfields, such as the Define Flexfields task list, Manage Sandboxes, and Highlight Flexfields for adding and editing flexfield segments.

Planning a descriptive flexfield can involve the following tasks:

1. Identify existing parameters.
2. Identify existing context values and whether the context value is derived.
3. Identify custom attributes and plan the descriptive flexfield segments, segment properties, and structure.
5. Plan initial values.
6. Plan attribute mapping to Oracle Business Intelligence objects.

**Identify Existing Descriptive Flexfield Parameters**

Some descriptive flexfields provide parameters that can be used to specify the initial value of a descriptive flexfield segment. The parameter is external reference data, such as a column value or a session variable. For example, if a flexfield has a user email parameter, you can configure the initial value for a customer email attribute to be derived from that parameter.

Review the list of available parameters in the Derivation Value field in the Create Segment page for a descriptive flexfield. If you decide to use one of the parameters to set an initial value, select that parameter from the Derivation Value drop-down list when you add the descriptive flexfield segment.

**Evaluate Whether the Context Value Is Derived**

The context value for a descriptive flexfield might have been preconfigured to be derived from an external reference. For example, if the context is Marriage Status, then the value might be derived from an attribute in the employee business object. When the context value is derived, you might need to take the derived values and their source into consideration in your plan.

To determine whether the context value is derived, access the Edit Descriptive Flexfield task to view the list of configured context values for the flexfield. The Derivation Value field in the Context Segment region displays a list of available parameters.

If context values have been preconfigured, see Oracle Fusion Applications Help for product-specific information about the use of those values.

**Plan the Segments, Segment Properties, and Structure**

Identify the custom attributes you need for a business object to determine the segments of the descriptive flexfield. Determine the segment properties such as the prompt, display type, or initial value.

The structure of the descriptive flexfield is determined by its global, context, and context-sensitive segments. Plan a global segment that captures an attribute for
every instance of the business object. Plan a context for segments that depend on
a condition of situation applying to a particular instance of the business object.
Plan context-sensitive segments to capture attributes that are relevant in the
context.

There is only one context segment available for descriptive flexfields. If you
have more than one group of custom attributes where you could use the context
segment, you will have to pick one group over the others, based on your
company’s needs and priorities, and add the other custom attributes as global
segments.

Plan Validation Rules

Define each segment’s validation rules and check if value sets exist for those
rules or you must create new ones. If you must create a value set, you can create
it either before configuring the flexfield or while creating or editing a segment.

When determining a segment’s validation rules, consider the following
questions:

• What is the data type - character, date, date and time, or number?
• Does the segment require any validation beyond data type and maximum
length?
• Should a character type value be restricted to digits, or are alphabetic
characters allowed?
• Should alphabetic characters automatically be changed to uppercase?
• Should numeric values be zero-filled?
• How many digits can follow the radix separator of a numeric value? In
base ten numerical systems the radix separator is decimal point.
• Does the value need to fall within a range?
• Should the value be selected from a list of valid values? If so, consider the
following questions:
  • Can you use an existing application table from which to obtain the list
of valid values, or do you need to create a custom list?
  • If you are using an existing table, do you need to limit the list of values
using a WHERE clause?
  • Does the list of valid values depend on the value in another flexfield
segment?
  • Is the list of valid values a subset of another flexfield segment’s list of
values?

Plan Initial Values

For every segment, list the constant value or SQL statement, if any, to use for the
initial value of the custom attribute.
Plan How Segments Map to Oracle Business Intelligence Objects

If a descriptive flexfield has been enabled for Oracle Business Intelligence, you can make it available for use in Oracle Business Intelligence applications. You can use segment labels to map segments to logical objects. Plan to map segments to logical objects before deploying the flexfield as a way to streamline the import into Oracle Business Intelligence.

Use the Manage Segment Labels page to view preconfigured segment labels. If a segment label doesn't exist for the logical object, then decide on a code, name, and description in preparation for adding that label. Choose a code, name, and description that will help end users select the correct label.

The mapping equalizes similar context-sensitive attributes that are from different contexts but are mapped to a single logical object. For information about objects in the logical model, see the "Working with Logical Tables, Joins, and Columns" chapter in the Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

Managing Descriptive Flexfields: Points to Consider

Configuring descriptive flexfields involves managing the available flexfields registered with your Oracle Fusion Applications database and configuring their flexfield-level properties, defining and managing descriptive flexfield contexts, and configuring global and context-sensitive segments.

Every descriptive flexfield is registered to include a context segment, which you may choose to use or not.

In general, configuring descriptive flexfields involves:

1. Creating segment labels for business intelligence enabled flexfields.
2. Configuring global segments by providing identity information, the initial default value, and the display properties.
3. Configuring the context segment by specifying the prompt, whether the context segment should be displayed, and whether a value is required.
4. Configuring contexts by specifying a context code, description, and name for each context value, and adding its context-sensitive segments, each of which is configured to include identifying information, the column assignment, the initial default value, and the display properties.

The following aspects are important in understanding descriptive flexfield management:

- Segments
- Adding Segments to a Highlighted Flexfield
- Usages
- Parameters
- Delimiters
• Initial Values
• Business Intelligence

Segments

You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order. You cannot enter a number for one segment that is already in use for a different segment.

Value sets are optional for context segments. The value set that you specify for a context segment consists of a set of context codes, each of which corresponds to a context that is appropriate for the descriptive flexfield. The value set must be independent or table-validated. If table-validated, the WHERE clause must not use the VALUESET.value_set_code or SEGMENT.segment_code bind variables. The value set must be of data type Character with the maximum length of values being stored no larger than the context's column length.

If you don’t specify a value set for a context segment, the valid values for that context segment are derived from the context codes. The definition of each context segment specifies the set of context-sensitive segments that can be presented when that context code is selected by the end user.

For reasons of data integrity, you cannot delete an existing context. Instead, you can disable the associated context value in its own value set by setting its end date to a date in the past.

You can configure the individual global segments and context-sensitive segments in a descriptive flexfield. These segment types are differentiated by their usage, but they are configured on application pages that use most of the same properties.

Adding Segments to a Highlighted Flexfield

When you highlight flexfields on a 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>
</tbody>
</table>
### Display Component | Value Set Created with Add Segment
--- | ---
List of Values | Independent
Radio Button Group | Independent
Text Field With Search | Independent
Text box | Format Only
Text area | Format Only
Date/Time | Format Only

---

**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 \(<\text{name}>\)_ and \(<\text{name}>\)_c attributes for each segment, along with some other optional attributes. The \(<\text{name}>\)_ attribute contains the value. The \(<\text{name}>\)_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 attributes in groups to form a context so that 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.

• Application development has registered some extensible flexfields to support view and edit privileges. For such flexfields, you can specify view and edit privileges at the context level to control who sees the attributes and who can change the attributes’ values.

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 it’s deployment status updated when the background deployment process has completed.
Note

The **Offline Status** column refreshes when you perform a new search in the Manage Extensible Flexfields task.

---

**Configuring an Item Extended Attributes Flexfield: Example**

The Item Extended Attributes flexfield provides segments for extending the Item business object. In the Manage Extensible Flexfields task, you configure your product business object to include a Technical Specifications logical page in the user interface for the Electronics and Computers category of items.

In this example, your configuration of this flexfield groups custom attributes into the following contexts:

- Materials and Substances
- Compliance and Certification
- Voltage

**Scenario**

The following list shows an example plan for custom computer attributes for the Item Extended Attributes flexfield. In this example, the Electronics Information page is inherited from the parent Electronics and Computers category.

- Page: Electronics Information
  - Context: Compliance and Certification, single row
    - ISO 14001 (International Organization for Standardization for an Environmental Management System)
    - ENERGY STAR (energy efficiency guidelines)
    - ROHS (Restriction of the use of certain hazardous substances in electrical and electronic equipment)
  - Context: Voltage, single row
    - Minimum voltage
    - Maximum voltage
    - Current type
  - Context: Materials and Substances, multiple rows
    - Material
    - Contain recyclate
    - Percent unit mass
- Page: Computer Information
  - Context: Processor Specifications, single row
- Manufacturer
- CPU type
- Processor interface
- Processor class
- Processor speed
- Cores

The following table summarizes key decisions for this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which extensible flexfield is available for configuring a hierarchy of categories?</td>
<td>Item Extended Attributes flexfield</td>
</tr>
</tbody>
</table>

**Collecting Technical Specifications**

Your product inventory pages for electronics and computers require a technical specifications page. Your product inventory pages for furniture require a furniture specifications page and an assembly instructions page. Items in both the electronics and computer category, and in the furniture category, share attributes for specifying materials and substances.

The figure shows a Technical Specifications logical page in the user interface for the Electronics and Computers category, with attributes in the context of Recovery and Recycling, Compliance and Certification, Operating Conditions, and Materials and Substances. The Materials and Substances context is configured for multiple rows so your users can select all the materials and substances required to make a single product, displayed as attribute values in a table.

**Analysis**

You use logical pages to arrange how the contexts appear in the user interface. Use a context to store all the materials and substances required to make a single
You can configure a context to store multiple rows per entity. The multiple rows are displayed in a table, as for the Materials and Substances context.

The Technical Specifications logical page contains the attributes for the four contexts.

- Recovery and Recycling
- Compliance and Certification
- Operating Conditions
- Materials and Substances

In the figure, the Furniture category is configured to include a Furniture Specifications logical page and an Assembly Instructions logical page. The two categories (Electronics & Computers and Furniture) share the Materials & Substances context.

Configure Security for the Item Flexfield Configuration

The following table shows an example of data security policies for the Item flexfield.

<table>
<thead>
<tr>
<th>Data Security Resource</th>
<th>Policy</th>
<th>Role</th>
<th>Actions</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM_EFF_B</td>
<td>A</td>
<td>VOLTAGE_SPEC</td>
<td>edit_nontrans_voltage_ctx</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>B</td>
<td>COMPLIANCE_SPEC</td>
<td>edit_trans_compliance_ctx</td>
<td>All values</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>C</td>
<td>COMPUTER_SPEC</td>
<td>edit_trans_attrs</td>
<td>ComputerCategoryFilter</td>
</tr>
<tr>
<td>ITEM_EFF_VL</td>
<td>D</td>
<td>TELEVISION_SPEC</td>
<td>edit_trans_attrs</td>
<td>TVCategoryFilter</td>
</tr>
</tbody>
</table>
The following table shows the privileges for three of the flexfield’s contexts.

<table>
<thead>
<tr>
<th>Context</th>
<th>Edit Privilege</th>
<th>View Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>edit_nontrans_voltage_ctx</td>
<td>NONE</td>
</tr>
<tr>
<td>Compliance and Certification</td>
<td>edit_trans_compliance_ctx</td>
<td>NONE</td>
</tr>
<tr>
<td>Materials and Substances</td>
<td>edit_trans_attrs</td>
<td>NONE</td>
</tr>
</tbody>
</table>

In this example, anyone can view the contexts’ attributes, but the edit privileges are restricted as follows:

- Voltage: Editable only by voltage specialists.
- Compliance and Certification: Editable only by compliance specialists.
- Materials and Substances: Only computer specialists can edit these attributes for items in the computer category. Only television specialists can edit these attributes for items in the TV category.

In this example, the Materials and Substances context is secured by a generic action with a condition applied to restrict access by category. Voltage and Compliance and Certification are secured by actions specific to each context.

**FAQs for Manage Extensible Flexfields**

**Why did the extensible flexfield context not appear at run time?**

If a deployed extensible flexfield context doesn’t appear in the user interface, verify that the context is associated with one of the category’s pages defined for the extensible flexfield.

**Manage Key Flexfields**

**Key Flexfields: Explained**

Key flexfields provide a means to capture a key such as a part number, a job code, or an account code. A key flexfield consists of one or more segments, where each segment can have a meaning.

For example, a part number 10-PEN-BLA-450 might correspond to a black pen from vendor #450 sold by division #10 (office supplies). Behind the scenes, the application uses a unique number, 13452, for this part, but the end user always sees the 10-PEN-BLA-450 part number.

The following aspects are important to understanding key flexfields:

- Architecture
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>Level Of Dynamic Combination Creation</td>
<td>Controlled By:</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------------------------------------------</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.

**Planning Key Flexfields: Points to Consider**

Your first step in planning your key flexfields is to determine which key flexfields your application requires.

Your plan should include:

- The purpose of the key flexfield
- The number and length of its available segment columns
- Whether your key flexfield allows more than one structure
- Whether more than one structure must be defined
- The number, order and length of your segments for each structure

Consider the following aspects in planning flexfields:

- Before you begin
- Access to flexfield-related tasks
- Restrictions
- Validation rules for flexfield segments

**Before You Begin**

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the
run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users.

Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list and Manage Sandboxes.

If you plan to use value sets, create them before configuring the key flexfield. You cannot create value sets for key flexfields at the time that you add and configure key flexfield segments.

**Access to Flexfield-Related Tasks**

To access tasks for configuring flexfields and value sets, you must be provisioned with roles that entitle you to access the tasks in the Define Flexfields task list or tasks for managing product-specific flexfields. Contact your security administrator for details. For information about product-specific flexfield tasks, such as Manage Fixed Assets Key Flexfields, consult the product-specific documentation in Oracle Fusion Applications Help.

**Restrictions**

If you plan to use value sets, create them before configuring the flexfield.

Plan your key flexfield configuration to scale to your enterprise needs. For example, if you expect to disable old cost centers and enable new ones frequently, plan a larger maximum size for your cost center value set so that you can have more available values. One thousand available values for a 3-character value set provides more room for changes than 100 available values for a 2-character value set.

Note the code name of the flexfield you intend to configure so you can find it easily in the Define Flexfield task list or tasks for managing product-specific key flexfields.

In some cases you can customize how the flexfield appears on the page.

See Oracle Fusion Applications Help for specific products to determine any restrictions on using product-specific key flexfields.

**Reporting**

If you want to report on your data by certain criteria or sub-entities, such as account number or project or region, consider making that sub-entity a distinct segment, rather than combining it with another sub-entity, so that you can categorize and report on smaller discrete units of information.

**Managing Key Flexfields: Points to Consider**

Consider the plans for a key flexfield, security, and resulting 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.

**Cross Validation Rules: Explained**

You can control the creation of new key flexfield code combinations by defining cross validation rules. A cross-validation rule defines validation across segments and enforces whether a value of a particular segment can be combined with specific values of other segments to form a new combination.

The table compares segment validation to cross-segment validation:

<table>
<thead>
<tr>
<th>Type of validation</th>
<th>Type of control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment validation</td>
<td>Controls the values you can enter for a particular segment</td>
</tr>
<tr>
<td>Cross-segment validation</td>
<td>Controls the combinations of values that administrators and end users can create for key flexfields</td>
</tr>
</tbody>
</table>

**Note**

You can use cross-validation rules for any key flexfield that has cross-validation enabled. See the documentation for your key flexfield to determine if it supports cross validation.

Cross-validation rules prevent the creation of combinations with values that shouldn’t coexist in the same combination. For example, your company requires
that all revenue accounts must have a specific department. Therefore, account combinations that have revenue account values, such as all values between 4000 and 5999, must have a corresponding department value other than 000, which indicates no department is specified. You can define cross validation rules that disallow creation of combinations with incompatible segments, such as 4100-000 or 5000-000.

Alternatively, suppose your accounting key flexfield has an Organization segment with two possible values, 01 and 02. You also have a Natural Account segment with many possible values, but company policy requires that Organization 01 uses the natural account values 001 to 499 and Organization 02 uses the natural account values 500 to 999. You can create cross-validation rules to ensure that users cannot create a general ledger account with combinations of values such as 02-342 or 01-750.

The following aspects are important to understanding cross validation rules:

- Rule Definitions
- Enforcement
- Timing

**Rule Definitions**

Cross validation rules consist of the following information:

<table>
<thead>
<tr>
<th>Rule Feature</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Uniquely identifies cross validation rules in a deployment.</td>
</tr>
<tr>
<td>Description</td>
<td>Helps administrator(s) identify the purpose of the rule.</td>
</tr>
<tr>
<td>Error message</td>
<td>Explains why the attempted combination violates the rule.</td>
</tr>
<tr>
<td>Start Date, End Date</td>
<td>Indicates the period of time when the rule is in effect.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Determines whether the rule is enforced.</td>
</tr>
<tr>
<td>Condition filter</td>
<td>Determines the conditions under which an enabled cross validation rule should be evaluated.</td>
</tr>
<tr>
<td>Validation filter</td>
<td>Determines the validation that the rule enforces when that condition is met.</td>
</tr>
</tbody>
</table>

When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created. If the event specified in the condition filter isn't applicable, then the combination is considered to pass the rule and the rule won't be evaluated even if it is enabled.

**Note**
If you don’t specify any statement in the condition filter, then the condition is always true and the rule is always evaluated.

**Enforcement**

Cross-validation prevents creation of invalid combinations by administrators using maintenance pages and end users using dynamic insertion in foreign key pages.

Enabled rules are enforced when there is an attempt to create a new combination of segment values. Disabled rules are ignored. Deleting the rule has the same effect, but you can re-enable a disabled rule.

**Timing**

When users attempt to create a new combination, the key flexfield evaluates any cross-validation rules that are enabled and in effect.

**Warning**

Cross-validation rules have no effect on combinations that already exist. The flexfield treats any existing invalid combinations that pre-date the rule as valid.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, manually disable those combinations using the combinations page for that key flexfield.

When defining a cross-validation rule, specify a start and end date to limit the time when the rule is in effect. The rule is valid for the time including the From and To dates.

**Cross Validation Rules: Points to Consider**

When you need key flexfield combinations of segment values validated across segments, you can optimize your cross validation rules to improve the experience of administrators and end users.

Consider the following when defining cross validation rules:

- Filters
- Rule Complexity
- Maintenance

**Filters**

A cross validation rule includes a condition filter and a validation filter.

The rule is evaluated using the following logic: If the condition filter is satisfied, then validate that the validation filter is satisfied.

1. The condition filter describes the event under which the rule will be evaluated. If the event specified in the condition filter isn’t applicable, then the rule won’t be evaluated even if it is enabled.

2. When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created.
For example, if your organization has determined that a certain company value, Operations, cannot use a specific cost center, Marketing, you can define a cross validation rule to validate your combinations.

1. The rule evaluates the company condition filter.
2. When company is equal to Operations, the rule evaluates the cost center validation filter.
3. When cost center is equal to Marketing, the rule prevents a combination from being created.
4. The error message you defined for the rule displays to inform the user that the attempted combination violates the rule.

Note
This rule doesn’t affect the creation of combinations with Marketing cost center and company values other than Operations.

Rule Complexity
For optimal performance and ease of understanding, define several simple validation rules instead of using one complex rule. Simple validation rules let you provide a more specific error message and are easier to maintain over time. Avoid rules that control validation across more than two segments, where possible. While you can define cross validation rules that span two or more segments, keep in mind that it becomes more difficult to interpret cross validation error messages and correct invalid key flexfield combinations as your rules encompass more segments.

Maintenance
To maintain consistent validation, review existing key flexfields when you update your cross validation rules. Regardless of your current validation rules, Oracle Fusion Applications accept a key flexfield combination if the combination already exists and is enabled. Therefore, to ensure accurate validation, you must review your existing combinations and disable any combinations that don’t match the criteria of your new rules.

Tip
To keep this type of key flexfield maintenance to a minimum, decide upon your cross validation rules when you first set up your key flexfield structure. Define cross validation rules before creating combinations and before combinations are used in transactions.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross validation rules, disable those combinations using the combinations page.

Creating a Cross Validation Rule: Example
Create cross validation rules to prevent specific combinations of segment values in your account combinations, for example, preventing a particular cost center
from being combined with a specific company value. Cross validation rules only affect the creation of new account combinations.

**Scenario**

Enter a new cross validation rule to prevent your InFusion America Inc. company value 01 from being combined with your marketing department value 300 in an account combination. Your company, InFusion America Inc. does not have a marketing department.

1. Navigate to the Manage Cross-Validation Rules task from within your implementation project, and then click the Go to Task icon.
2. Select your InFusion America chart of accounts.
3. Click the Create icon.
4. Specify a unique rule Name, IFAM01, and an optional Description, Do not combine Marketing Department, 300 with InFusion America, company 01.
5. Enter an optional effective From Date of today. Check Enabled.
6. Click the Change filter condition on the Condition Filter. Enter Company equal to 01. The cross validation rule evaluates if Company 01 was entered and if it was entered, then the validation process continues to evaluate the rule.

**Note**

If you do not specify any statement in the condition filter, then the rule is always evaluated.

7. Click on the Change filter condition on the Validation Filter. Enter Cost Center is not equal to 300. When the rule is evaluated, an account combination must contain a cost center other than 300 before it can be created.
8. Enter an Error Message: Cost Center 300 is not allowed with Company 01. The message displays in the relevant user interfaces and processes when an account combination cannot be created because it violates the rule.
9. Click Save and Close.

**Enabling Key Flexfield Segments for Business Intelligence: Points to Consider**

A key flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segment instances. When a segment instance is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled key flexfield segments.

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component
• Mapping attributes of flattened business components to logical objects in Oracle BI
• Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.

Flattening

When you deploy a business intelligence-enabled key flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn't assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

Mapping to Logical Objects in Business Intelligence

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence.

If you assign a label to segments that serve the same purpose in different structures, you can consolidate or equalize the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, an organization may have more than one definition of its key accounting flexfield to support different requirements for accounting reporting, or due to chart of accounts definitions from acquired organizations. A US Accounting Flexfield structure may have a segment called Subaccount to track project expenditures. The same type of information may be tracked in a UK accounting flexfield structure with a segment called Project. Equalize these two segments to create a single list of values for reporting.

Non-labeled segments aren’t equalized across context values, so the flattened components include a separate attribute for each segment for each structure.

Note

It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

Note

Assigning a label to a segment serves to equalize the attribute across structures, as well as map the equalized attribute to business intelligence.
Managing Labels

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify 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> and <name>_c attributes for each segment, along with some other optional attributes. The <name> attribute contains the value. The <name>_c attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

Key Flexfields: Example

A key flexfield can capture expense account information.

Scenario

When entering details for each expense, the user specifies an account to which the expense is charged.
Enter Expense Accounts

A user interface for entering expenses gives the user the option of selecting an expense account that identifies the cost center and other details needed for processing the expense.

Analysis

The expense account field is a foreign key reference to a code combination (EXPENSE_LINES.EXPENSE_ACCOUNT = ACCOUNTS.CCID).

Code Combinations Table for Entering Accounts and Employees

The code combinations table supports entering account information, such as for expense accounts.

The figure shows the origin in the code combinations table of the account specified by the user. The code combination ID record stores the information of the key flexfield segments used to assemble the expense account based on the key flexfield configuration.

The combinations page, which is the maintenance page for the key flexfield, is for managing rows in the combinations table. In this example, managing the combinations means adding or editing account numbers that adhere to the key flexfield metadata rules.
The figure shows the code combination details for the example expense account reflected in the flexfield configuration and the code combinations table.

If dynamic combination creation isn’t enabled, then when entering an expense line, the user can only select an account that already exists in the ACCOUNTS (combinations) table. If they require an account that doesn’t exist, they must consult with the appropriate application administrator who can add the account to the combinations table.

If dynamic combination creation is enabled, then when entering an expense line, the user can either select a pre-existing account, or type in a new account that is created dynamically on the fly in the ACCOUNTS (combinations) table. Once the new combination is created, the same user can refer to it on the expense line.

When managing employee information, the user specifies the cost center that the employee belongs to. The cost center field corresponds to a single, labeled segment of the Account Key Flexfield and has metadata defined such as the allowable value set for that segment.

In this figure, instead of specifying a cost center ID reference to an account, only the Cost Center segment is used and the value is stored directly on the employee table.
Define Attachments

Attachments: Explained

Attachments are pieces of supplementary information that users can associate with specific business objects such as expense reports or purchase orders. Attachments can be URLs, desktop files, text, or in cases where available, repository folders. For any given business object, a user may be able to only view attachments, or also create, delete, or edit attachments, depending on security. For more information on an introduction to attachments, see the Oracle Fusion Applications Developer’s Guide.

Repository

Attachments are stored in a content management repository provided by Oracle WebCenter Content Server. Users managing attachments have no real interaction with the repository unless the repository mode is enabled for attachments on specific business objects. In that case, users can share attachments among objects, update attachments by checking them out of and back into the repository, and perform other tasks. Access to attachment files is controlled by a digital signing mechanism. Depending on security, users might have direct access to the repository.

Security

Data security that applies to a specific business object also applies to attachments for that object, as determined by the attachment entity defined for the object. For example, if a user has no access to a specific expense report, then the same user cannot access attachments for the expense report. You can also use attachment categories to control access and actions on attachments, based on roles associated with the category. For more information on securing attachments, see the Oracle Fusion Applications Developer’s Guide.

Attachment Entities: Explained

An attachment entity is usually a database entity, for example a table or view, that represents a business object attachments can be associated with. Each attachment UI must be defined with a corresponding attachment entity, which not only identifies the business object to attach to, but also controls what users can do. Attachment entities are used only in the context of attachments and exist separately from the database entities that they are based on.

Edit and create attachment entities on the Manage Attachment Entities page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Attachment Entities task. Though you would
generally use predefined attachment entities with attachment UIs, you might need to create new entities, for example when developing custom UIs.

**Entity Names**

An attachment entity name should match the name of the table or view that represents the business object to attach to. The name is also used in the repository folder that is automatically created to store attachments for the entity. The attachment entity display name should be something that users know to represent the business object.

**Database Resource**

The data security policies associated with the database resource defined for the attachment entity would apply to attachments for that entity. For example, based on the database resource for the expense reports attachment entity, the same policies apply to attachments for expense reports. The database resource value must match the value in the OBJ_NAME column in the FND_OBJECTS table for the business object that the entity represents.

**Enabling Security**

Security based on the database resource associated with the attachment entity is always in effect. What you can enable or disable is security based on attachment categories. If any of the attachment categories associated with the attachment entity has data security defined, then that security applies to this entity only if enabled.

**Attachment Entities and Attachment Categories: How They Work Together**

The association between attachment entities and categories determines which categories can be used for an entity. For example, categories associated with the expense report attachment entity are available to be implemented in attachment UIs for expense reports. You can define these associations when managing either entities or categories. Any association changes in either the Manage Attachment Entities or Manage Attachment Categories page are reflected on the other page. You can access either page by starting in the Setup and Maintenance Overview page and searching for attachment tasks.

**Managing Entities**

You determine which attachment categories are relevant to a particular entity on the Manage Attachment Entities page, and each entity must have at least one category. Depending on configuration, any or all of the available categories for that entity are used. For example, you assign three categories to the expense reports attachment entity. For a particular expense report page with attachments functionality, you can customize the attachments component to specify which of the three categories are used. Based on your selection, the data security defined for each category, if any, is applied to attachments on that page if the attachment entity has category-based security enabled.
Managing Categories

If you create an attachment category and need to assign it to multiple attachment entities, use the Manage Attachment Categories page. The association means the same as the association on the Manage Attachment Entities page.

Attachments Troubleshooting: Explained

Attachments UIs for users to add and manage attachments are fully functional as is, and users usually would not encounter issues. If you customize attachments in any way, for example by creating additional attachment categories and implementing data security on them, then some issues might arise.

Issue: Unable to View, Add, Update, or Delete Attachments

Users encounter issues when trying to view attachments or perform actions such as adding attachments.

- Users can no longer see specific attachments that they were previously able to see.
- Likewise, they can no longer update or delete attachments.
- Users get an error stating that they do not have permission to add attachments.

Resolution

Use the Manage Attachment Entities page to ensure that attachment categories are associated to the relevant attachment entity. For example, if users can no longer see attachments for an expense report, then search for the expense report attachment entity and assign all necessary categories to it. You might need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or what categories to assign.

If data security is implemented on the categories for the attachment entity, then verify that the Enable Security check box is selected in the Manage Attachment Entities page for that entity. Make sure that users have a role with the privileges shown in the following table, to view, add, update, or delete attachments with a specific attachment category.

<table>
<thead>
<tr>
<th>Action</th>
<th>Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>Read Application Attachment (FND_READ_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
<tr>
<td>Add or Update</td>
<td>Update Application Attachment (FND_UPDATE_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete Application Attachment (FND_DELETE_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
</tbody>
</table>
For example, if users have the Read Application Attachment privilege for all categories associated with the expense report attachment entity, except the Receipts attachment category, then they can view all expense report attachments except those created with the Receipts category. Likewise, if users do not have the Update Application Attachment privilege for any attachment categories tied to the expense report attachment entity, then they cannot create any attachments at all for expense reports.

For more information on attachment category data security, see the Oracle Fusion Applications Developer’s Guide.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

**Issue: Missing Attachment Category**

Users can see existing attachments, but the attachments no longer have an attachment category value.

**Resolution**

When the attachment was added, at least one category existed for the corresponding attachment entity, as otherwise the attachment could not have been added. Since then, the entity was edited so that it no longer has any assigned categories, so the user cannot see the category associated with that attachment.

Use the Manage Attachment Entities page to reassign attachment categories to the relevant attachment entity. For example, if users can no longer see the Receipts attachment category for an attachment to an expense report, then search for the expense report attachment entity and assign to it the Receipts category. You might need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or what additional categories to assign.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

**FAQs for Define Attachments**

**What’s an attachment category?**

An attachment category is used to classify and secure attachments. Each attachment user interface must be defined with at least one category for users to be able to add attachments. If there are multiple categories, users can view...
them and select one when adding attachments. For example, attachments for an expense report can be categorized as receipts, scanned invoice images, and so on.

You can also associate roles with categories to determine user access and actions for attachments, based on the categories assigned to the attachment entity. For example, security for expense report attachments can be based in part on the categories assigned to the expense report attachment entity. You can define multiple categories per module, and add and manage custom categories for your own purposes. For more information on attachment category data security, see the Oracle Fusion Applications Developer’s Guide.

Use the Manage Attachment Categories page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Attachment Categories task.
Source Systems: Explained

You can set up source systems to enable users to identify the source of the data they are importing. You can specify whether the source system is a spoke system, such as a legacy system, or a purchased system, such as data from a third party provider. You can also specify what types of entities the source system contains. For example, you can specify that a source system will contain customer data.

You can configure the following for a source system:

- Source system code, name, and description
- Source system type
- Enable for Items, Trading Community Members, Order Orchestration and Planning, and Assets

Source System Code, Name, and Description

You can create a source system code to uniquely identify the source system. Source system codes are used by the application to create references between source IDs and the Oracle Sales Cloud database IDs.

Note

You cannot update the source system code once you have created the source system.

Source System Type

You must set up a source system as either a spoke system, such as a legacy system, or a purchased system, such as data from Dun & Bradstreet.
Enable for Items, Trading Community Members, Order Orchestration and Planning, and Assets

You should select which types of entities will be imported from the source system into the Oracle Sales Cloud database from the following:

- Items
- Trading Community Members
- Order Orchestration and Planning
- Assets

You can select one or more of these entity types as required for the source system. It is important to enable the correct entity types because each import UI filters source systems based on their entity type. For example, if a source system is enabled for Trading Community Members, Items, and Assets, then the source system can be selected as a data source in the Trading Community Members, Items, and Asset import UIs; however, the source system won’t be able to be selected in the Orchestration and Planning import UI.

Source System Entities: Explained

Source System Entities are the entities, such as addresses and parties, which can be imported using a specified source system.

When you import data from a source system, all of the entities in the source system data will be imported. Within the Source System Entities UI, you can chose to allow multiple source references, which allows multiple records from a source system to map to a single record.

FAQs for Define Source Systems

What happens if I allow multiple source system references?

Allowing multiple source system references means that when you import data from a source system you can merge multiple, or duplicate, source system records and create one record in the Oracle Sales Cloud.

If you do not allow multiple source system references then a record in Oracle Sales Cloud will be created for every source system record. This means that you could potentially create duplicate records in Oracle Sales Cloud.
Common Customer Data Management Configuration: Define Party Usages

Party Usage: Explained

Party usages describe how a party is used in the context of the implementing organization. For example, a person in the business community may be a consumer or contact. Rules can be associated to a party, and these rules determine when a party usage can and cannot be used.

These are the following rules that can be defined for a party usage:

• Assignment rules
• Exclusivity rules
• Incompatibility rules
• Transition rules

Assignment rules

Assignment rules are used to define how the party usage can and cannot be assigned to parties. You can choose to allow unconditional party usage assignment, enabling the assignment to be manually created, or updated, by an administrator.

Alternatively, you can restrict the manual assignment and update of a party usage, which will mean that the party usage can only be created, or updated, by a business event. For example, creating an account for a party record can automatically assign the Customer party usage to that record.

You can also specify whether the party usage assignment can also be created or updated by the assignment of a party relationship. For example, the party relationship Contact Of can assign a party usage of Organization Contact.

Exclusivity rules

Exclusivity rules enable you to restrict party usage assignment, so that between a specified date range the party usage can be the only usage allowed to be
assigned to a party. For example, you can set up an exclusivity rule that between January 1 2011 and February 1 2011 parties that have the Manufacturer party usage cannot have any other party usage assigned to them.

**Incompatibility rules**

You can specify which party usages cannot be assigned concurrently to a party between a defined date range. For example, you can specify that you cannot assign a Sales Account usage to a party with a Sales Prospect usage, during January 1 2011 and December 31 2030.

**Transition rules**

You are able to define which party usages can transition to the party usage you are creating or editing. Once a party has transitioned to the current party usage, the previous party usage is set with an end date. For example, when a party with the Prospective Partner party usage is assigned the Partner party usage, the Prospective Partner usage is set with an end date.

**Party Usage Filter Rules: Explained**

Party usage filter rules are a grouping of party usages that are used for searching and filtering in user interfaces. You can specify which party usages you want to include and exclude, and whether the party usage assignments are active or inactive.

For example, an Eligible Customers filter rule can include the Sales Prospect party usage, Sales Account, Legal Entity, and Customer party usages. The Eligible Customers filter rule can then be used in transactional UIs to filter out parties with party usages that are not included in the rule.

---

**Note**

Once a party usage filter rule is created you need to compile the filter rule before you can use the filter rule in a User Interface.

---

**FAQs for Define Party Usages**

**How can I compile a party usage filter rule?**

In the Party Usage Filter Rule page click **Actions** in the header section, and then click **Compile Filter Rule**.
Add additional name types to capture alternative names for parties.

For example, create an additional name type NICK_NAME to capture the nick names of parties of type Person.

While creating an additional name type, specify the party type to which it applies. From then on, the additional name type will be available as a naming option for all parties of that party type within the deploying company. You can use additional name types to capture language-specific names as well.
Identifier Types: Explained

Create additional identifier types to provide extensions to party attributes. For example, you can create an additional identifier type to record a person's passport number.

You can choose which party types can use the additional identifier type. These party types can be Person or Organization, or both. You can also specify whether the value of an identifier type must be unique. For example, the passport number listed under each person's profile must be unique.

Additional identifier types do not automatically appear in the user interface. If you want to use identifier types in the application you will need to call the Trading Community Member Name and Identifier Setup web service.
Common Customer Data Management Configuration: Define Relationships

Relationships Types: Explained

A relationship type categorizes relationship phrases and roles that determine the nature and purpose of a relationship. A relationship type defines the aspects of a relationship such as the roles of the subject and object, business rules, and phrases that describe the relationship. A relationship, uses the relationship type, to define how a party is related to another party.

The components of a relationship type are:

- Relationship phrase pair and role pair
- Relationship characteristics

**Relationship phrase pair and role pair**

A relationship type contains a relationship phrase pair and a role pair. Relationship phrase pair and role pair define the mutual roles that the two parties play in a relationship. For example, for relationship type Employment the phrase pair contains Employee of and Employer of; and, the role pair contains Employee and Employer.

The role pair describes the roles of the entities immaterial of the direction of the relationship. For example, in an Employment relationship Bill has the role of an employee and Oracle Corporation has the role of an employer.

The phrase pair describes the roles of the entities in the relationship. For example, in an Employment relationship if Bill is the subject and Oracle is the object, then the phrase would be Employee of. Similarly, if Oracle is the subject and Bill is the object, then the phrase would be Employer of.

**Relationship characteristics**

Relationship types include characteristics or rules that define how the relationship type can be used.
While defining a relationship type, you can define if the relationship can be:

- Circular
- Related to self

Circular relationships have two parties related in a circle of relationships. For example, you can create a Competitor relationship type to link two organization parties. In this scenario, ABC Corp is a competitor to XYZ Corp, and XYZ Corp is also a competitor to ABC Corp.

You can create a self-relating relationship type, where the subject and the object of the relationship are the same. For example, you can create a Subsidiary relationship type, where a subsidiary organization ABC is linked to a parent organization XYZ. The parent organization XYZ, in turn, could be linked to itself as a subsidiary using the same relationship type.

### Relationship Groups: Explained

Relationship groups categorize relationship phrase and role pairs for functional purpose. For example, you can assign the relationship phrase Subsidiary Of to a group, and assign the phrase Headquarters Of to a different group.

#### Uses of Relationship Groups

Relationship groups are used to determine which relationship roles and phrases are displayed in specific user interfaces of the Oracle applications. A relationship group can correspond to one or more Oracle Applications user interfaces. For example, you can configure an Oracle application to display only the relationship types that pertain to the Party Contacts information.

### Manage Relationship Types

#### Creating Relationship Types: Worked Example

This example demonstrates how to create a relationship type, and add relationship roles and phrases to the relationship type.

The following table summarizes key decisions while relationship types.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What are the subject and object party types for the relationship?</td>
<td>The subject type and object type is organization.</td>
</tr>
<tr>
<td>Is the relationship circular?</td>
<td>Yes</td>
</tr>
<tr>
<td>Can the subject related to self in this relationship?</td>
<td>No</td>
</tr>
</tbody>
</table>
You must search for the relationship type to make sure it does not already exist, before you create a relationship type.

**Searching for a relationship type**

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

2. On the Manage Relationship Types page, in the Search region, complete the fields, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship Type</td>
<td>Equals</td>
<td>Vendor</td>
</tr>
<tr>
<td>Subject Type</td>
<td>Equals</td>
<td>Organization</td>
</tr>
<tr>
<td>Object Type</td>
<td>Equals</td>
<td>Organization</td>
</tr>
</tbody>
</table>

3. Click Search.

Review the search results. Ensure that there is no existing Relationship Type for the criteria.

**Creating a relationship type**

1. In the Search Results region, click the Create icon to open the Create Relationship Type page.

2. On the Create Relationship Type page, in the Overview region, enter Vendor in the Relationship Type field.

3. Select the Circular check box.

4. In the Relationship Roles and Phrases region, click the Create icon to open the Create Relationship Role and Phrase Pair dialog box. When you create a relationship with a relationship phrase or role, the reverse direction is automatically created with the other phrase or role in the pair.

5. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Type</td>
<td>Organization</td>
</tr>
<tr>
<td>Subject Phrase</td>
<td>Vendor of</td>
</tr>
<tr>
<td>Subject Role Singular</td>
<td>Vendor</td>
</tr>
<tr>
<td>Subject Role Plural</td>
<td>Vendors</td>
</tr>
<tr>
<td>Object Type</td>
<td>Organization</td>
</tr>
</tbody>
</table>
6. Click **OK**.

7. In the Create Relationship page, click **Save and Close**.

### Editing Relationship Types: Worked Example

This example demonstrates how to edit a relationship type, including adding relationship roles and phrases to the relationship type, and assigning the relationship to a group.

In this example, you will add a role and phrase pair to the relationship type Employment. You will also add the Employment relationship type to the Party Contacts group.

#### Editing a relationship type

1. From the Setup and Maintenance work area, search for the Manage Relationship Types task and click **Go to Task**.

2. On the Manage Relationship Types page, in the Search region, enter Employment in the **Relationship Type** field.

3. Click **Search**.

4. In the Search Results region, select the Employment relationship type.

5. Click the **Edit** icon to open the Edit Relationship Type page.

6. On the Edit Relationship Type page, in the Relationship Roles and Phrases region, click the **Create** icon to open the Create Relationship Role and Phrase Pair dialog box.

7. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Type</td>
<td>Organization</td>
</tr>
<tr>
<td>Subject Phrase</td>
<td>Has contract employees</td>
</tr>
<tr>
<td>Subject Role Singular</td>
<td>Contractor</td>
</tr>
<tr>
<td>Subject Role Plural</td>
<td>Contractors</td>
</tr>
</tbody>
</table>
8. Click **OK**.

9. In the Relationship Groups region, click the **Add Row** icon.

10. In the **Name** drop-down list, select **Party Contacts**.

11. Click **Save and Close**.

---

### Manage Relationship Groups

#### Creating Relationship Groups: Worked Example

This example demonstrates how to create a relationship group. Relationship groups are used to categorize relationship roles and phrases, based on the user interfaces they appear in.

In this scenario, you will create group for relationship types that link an organization to an external organization.

**Creating a relationship group**

1. From the Setup and Maintenance work area, search for the Manage Relationship Groups task and click **Go to Task**.

2. On the Manage Relationship Groups page, in the Search Results region, click the **Create** icon to open the Create Relationship Group page.

3. Complete the fields, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Related External Organizations</td>
</tr>
<tr>
<td>Code</td>
<td>REL_EXT_ORGS</td>
</tr>
<tr>
<td>Description</td>
<td>Indicates a group of relationships that link the organization to external entities.</td>
</tr>
</tbody>
</table>

---
4. In the Relationship Roles and Phrases region, click on the Add Row icon to add a relationship role and phrase pair.

5. For each relationship role and phrase pair, complete the fields, as shown in this table.

<table>
<thead>
<tr>
<th>Subject Type</th>
<th>Subject Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Supplier</td>
</tr>
<tr>
<td>Organization</td>
<td>Vendor</td>
</tr>
<tr>
<td>Organization</td>
<td>Competitor</td>
</tr>
</tbody>
</table>

6. Click Save and Close.
The classifications model provides you with a flexible tool to categorize entities such as parties, projects, tasks, and orders. Classifications enable you to classify an entity, such as a party, in a way that the rest of the world sees it, in addition to the way that it is referenced within your organization.

The major components of classifications are:

- Classification categories
- Classification rules
- Classification codes
- Classification code hierarchy
- Entity assignment

**Classification Categories**

Classification categories give you the ability to classify entities under a broad subject area. For example, you can classify organizations based on the industries they operate in. Classification categories are a logical grouping of one or more classification codes and allow classification code rules to be defined.

**Classification Rules**

Classification categories can have rules that define how classifications can be assigned to entities. When you set up classification categories specific rules can be created, such as allowing the parent classification code to be assigned to a party, and allowing multiple classification codes to be assigned to an entity.

**Classification Codes**

The individual values within the classification category are called classification codes. For example, in the 1987 SIC classification category there is a classification code of software that can be assigned to a party in the software industry. You can
organize classification codes into a hierarchical tree, with a parent classification code at the top of the tree and child classification codes branching off from the parent code or other classification codes.

**Classification Code Hierarchy**

You can create hierarchies of classification codes within a classification category. For example, you can set up a classification category of IT containing the classification codes hardware, keyboards, and printers. You can then set up the classification code of hardware as the parent code at the top of the tree, with the classification codes of keyboards and printers as child codes underneath. You can create further child classification codes, such as dot matrix, ink-jet, and laser below the printer classification code.

**Entity Assignment**

Define which entities can be assigned to a classification category by entering the entity table name and creating a Where clause in SQL. Only entities that satisfy the Where clause are assigned the classification category. For example, a classification category called industries with the Where clause of where "party_type = ORGANIZATION" would have the result that only organizations can be classified with the industries classification category.

**FAQs for Define Classifications**

**What happens if I allow parent code assignment?**

You can assign the parent classification code to an object, as well as the child classification codes. The parent classification code is the code at the top of the classification code tree.

If you don't allow parent classification codes to be assigned to an object, then you can assign only child classification codes, or codes that are below another classification code in the tree, to an object.

**What happens if I allow multiple class code assignment?**

You can assign more than one classification code from this classification category to an object.

If you don't allow multiple classification codes to be assigned to an object, then you can assign only one classification code from this classification category to an object.

**Can I update an existing entity assignment rule?**

No. You can delete the entity assignment rule and create a new one.
Data Formats help you define address formats and name formats within your organization, and assign them to specific geographies. These data formats can be used to capture and validate party name and address information. In order to use geography-based validations for addresses, you must set up geography data in addition to the address formats.

**Managing Data Styles**

You can define new styles for name and address, which would categorize a name and address style format. A Data Style, such as Mailing Address or Concatenated Name, categorizes name and address style formats. Oracle Sales Cloud uses Data Styles to identify a particular manner or situation for which an entity, such as name or address, is formatted. The application can be customized so that, for example, only certain name and address styles are displayed in the user interface.

**How Data Formats ensure Data Quality**

Use Data Formats to ensure party address and party name data quality. The format is used to present the address elements in the country specific formatting, and help users during address entry. This also provides the flexibility to specify the scope of address validation. Address formats are linked to the geographic data in order to provide address verification, at the data-import level. For example, when creating or editing an address for a party, selecting a country displays the address format for that country.

The use of Name formats ensures that name components are stored in a decomposed manner. This allows you to construct name representations from components in a variety of different styles. Name formats also allow you to store phonetic names that are required for certain geographies.

Data Formats simplify the storage of party address and party name, by storing them as components. Once you setup the address and name format for a geography, you can use the format to validate address and name data.
Setup Formats: Explained

Data Formats determine how names and addresses are displayed in the Oracle Sales Cloud. Name and address formats vary depending on usage, country, and language. You can create name formats and address formats, customize them to display the address and name components, and assign them to specific locales. This allows you to create data formats for various countries, languages, and usages. Additionally, you can create variations of the formats, with each variation having its own layout. For example, an address format may have variations of a general address, rural address, and military address.

There are two types of data formats.
- Address format
- Name format

Address Formats

Address formats specify the layout of an address, including the address components and their position in the layout. For example, an address format for US postal address can include address, city, state, and zip code. Similarly, the address format for UK would include address, city, county, and post code.

Name Formats

Name formats, such as Informal or Mailing, allow you to specify how a name is displayed in the application, based on the usage, language, and country. A name format includes name elements, their position, and formatting. You can create a name style format for either an organization name or a person name.

Creating an Address Style Format: Worked Example

This example shows how to create an address style format for a specified address style.

Specify the address style format name and address style, create the format variation layout, and assign a locale.

Creating the address style format name and specifying the address style

Create the address style format code and name, choose which address style you want to use to define this address style format, and specify if you want this address style format to be the default format for the selected address style.

1. On the Manage Address Formats page, click Actions and then click Create.

2. On the Create Address Style Format page, complete the fields in the Overview section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>CA_POSTAL_ADDR</td>
</tr>
</tbody>
</table>
3. Click Save.

Creating the format variation layout

Specify the address lines you want in the address and the position of the lines in the address layout. You also want to specify if the address lines will be mandatory, whether the address lines will render in uppercase, and whether there will be blank lines after the address line.

1. In the Format section on the Create Address Style Format page, click on the Format Layout tab.
2. In the Format Variation section, click Actions and then click New.
3. Enter a variation number in the Variation field.
4. In the Format Variation Layout section, click Actions and then click New.
5. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>1</td>
</tr>
<tr>
<td>Position</td>
<td>1</td>
</tr>
<tr>
<td>Prompt</td>
<td>Address line 1</td>
</tr>
<tr>
<td>Address Element</td>
<td>Address line 1</td>
</tr>
<tr>
<td>Required</td>
<td>Yes</td>
</tr>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

6. Click Actions, then click New to create another address line.
7. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>2</td>
</tr>
<tr>
<td>Position</td>
<td>2</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Prompt</td>
<td>City</td>
</tr>
<tr>
<td>Address Element</td>
<td>City</td>
</tr>
<tr>
<td>Required</td>
<td>Yes</td>
</tr>
<tr>
<td>Uppercase</td>
<td>Yes</td>
</tr>
</tbody>
</table>

8. Click **Expand** on the City address line, then enter the value 1 in **Blank Lines Before**.

9. Continue to create as many address lines as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, an address line. In this section you can also create an Attribute Transform Function to change a data value into a different value.

**Assigning a locale**

Specify the relevant countries for this address style format.

1. In the Format section on the Create Address Style Format page, click on the **Locale Assignment** tab.

2. Click **Actions** and then click **New**.

3. Choose a country for the address style format.

4. Continue to add the countries that will use this address style format.

5. Click **Save and Close**.

**Creating a Name Style Format: Worked Example**

This example shows how to create a name style format for a specified name style.

Specify the name of the name style format and name style, create the format variation layout, and assign a locale.

**Creating the name of the name style format and specifying the name style**

You create the name style format code and name, and choose which name style you want to use to define this name style format. You also specify if you want this name style format to be the default format for the selected name style.

1. On the Manage Name Formats page, click **Actions** and then click **Create**.

2. On the Create Name Style Format page, complete the fields in the Overview section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>CONCAT_NAME_US</td>
</tr>
</tbody>
</table>
3. Click Save.

Creating the format variation layout

Specify the name parts you want in the name, and the position of the name parts in the name layout. You also want to specify if the name parts will be mandatory, define whether the name parts will render in uppercase, and whether there will be blank lines after the name part.

1. In the Format section on the Create Name Style Format page, click on the Format Layout tab.
2. In the Format Variation section, click Actions and then click Add Row.
3. Enter a variation number in the Variation field.
4. In the Format Variation Layout section, click Actions and then click Create.
5. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>1</td>
</tr>
<tr>
<td>Position</td>
<td>1</td>
</tr>
<tr>
<td>Prompt</td>
<td>Person First Name</td>
</tr>
<tr>
<td>Name Element</td>
<td>First Name</td>
</tr>
<tr>
<td>Required</td>
<td>No</td>
</tr>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

6. Click Actions, then click Create to create another address line.
7. Complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>2</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Position</td>
<td>2</td>
</tr>
<tr>
<td>Prompt</td>
<td>Person Last Name</td>
</tr>
<tr>
<td>Name Element</td>
<td>Last Name</td>
</tr>
<tr>
<td>Required</td>
<td>No</td>
</tr>
<tr>
<td>Uppercase</td>
<td>No</td>
</tr>
</tbody>
</table>

8. Continue to create as many name parts as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, a name part. In this section you can also create an Attribute Transform Function to change a data value into a different value.

**Assigning a locale**

Specify the relevant countries for this name style format.

1. In the Format section on the Create Name Style Format page, click on the **Locale Assignment** tab.
2. Click **Actions** and then click **Create**.
3. Choose United States in the **Country** field.
4. Choose American English in the **Language** field.
5. Click **Save and Close**.

**FAQs for Define Data Formats**

**How can I create an address style format layout?**

Click **New** in the Manage Address Formats page and add address lines in the Format Variation Layout section. If you require more than one address style format layout, then you can create a format variation and add a different address format layout.

**How can I create a variation of an Address Style Format?**

You can create different variations of an address style format, with each variation having its own layout. For example, a country’s postal address may have variations such as general postal address, rural address, and military address. When you create an address style format, the layout defined for the format is
assigned as the default variation. Click Add in the Format Variation region of the Edit Address Style Format page, for the address style format, to add a variation of the Address style. For each address style format variation, you can enter a variation rank to define the priority of the variation.

How can I create a name style format layout?

Click New in the Manage Name Formats page and add line entries for each part of the full name in the Format Variation Layout section, within the Create Name Style Format page. If you require more than one name style format layout, then you can create a format variation and add a different name format layout.

How can I create another variation of a Name Style Format?

You can create different variations of a name style format, with each variation having its own layout. For example, different regions in a country may have different name style formats. Click Add in the Format Variation region in the Edit Name Style Format page, for the name style format, to add a variation of the name style. For each name style format variation, you can enter a variation rank to define the priority of the variation.
Telephone Numbering Plan: Explained

A telephone numbering plan defines the expected number and pattern of digits for a country phone number. It defines the country-specific components of phone numbers, display format styles, and codes. This includes country codes, international prefixes, area codes within a country, and mobile prefixes. You can specify country-specific mobile prefixes to determine if a phone number is mobile or not.

You can also create multiple telephone number display formats for a country. For example, for the US phone format, you define the phone country code as 01, fixed area code as three numbers in length, and subscriber number as seven digits in length. You can then create a telephone format for domestic display, such as (999) 999-9999. You can also create a telephone format for international display, such as 999-999-9999.

Updating Telephone Numbering Plans: Explained

You can update telephone number plans from the Edit Telephone Number Plan page. You cannot edit the country information if the country has area codes, mobile prefixes, or phone formats defined.

In the Edit Telephone Number Plan page, you can:

- Edit Scheme
- Edit Regional Information

Editing Scheme

You can edit the country-specific phone information in the Scheme section. You can edit the following fields in the section:
• Trunk Prefix: The code to dial before long distance numbers within the country, for example 1 for the US

• International Prefix: The code to dial before international numbers when calling from the country that you are setting up, for example 011 for the US

• Subscriber Number Length: The length of the subscriber number for countries where the subscriber length is fixed.

• Area Code Length: The length of the area code for countries where the area code length is fixed.

Editing Regional Information

You can edit the regional phone number information for the country in the Regional Information section.

The Regional Information section contains the following tabs:

• Area Codes: Allows you to define area codes that you want to use for the country.

• Mobile Prefixes: Allows you to define mobile prefixes that you want to use for this country. Mobile prefixes contain the area code and possible prefixes of the subscriber number. For example, the mobile prefix of 650506 consists of the 650 area code and the 506 subscriber number prefix.

• Telephone Number Formats: Allows you to define multiple telephone number display formats for a country.

How Setup Telephone Numbering Plans Fits in with the Other Setup Tasks

Telephone Numbering Plans allow you to define the telephone number format for a country. You can have multiple formats for a country, depending on the requirements.

How Telephone Number Plans Setup Fits in with other Setup Tasks

Phone number format, along with the other country-specific location information such as address and name formats, defines geographic-specific data formats. The Manage Telephone Numbering Plans page allows you to create and manage telephone number plans. Oracle Sales Cloud uses the telephone number formats to display phone number in the appropriate style, and to validate phone numbers. You can define country-specific components of phone numbers, display format styles, and area codes. Oracle Sales Cloud uses the phone formats to correctly parse phone numbers and determine the correct format style for displaying phone numbers in user interfaces.

Oracle Sales Cloud validates phone numbers for a country, based on the format defined for the country. The phone numbers are also validated against a common
set of rules that apply to all countries. You can also enter time zone information for a country, at the country code and area code level. The time zone information entered with phone number is validated against the time information defined for the country.

FAQs for Define Contact Points

Why can't I update the country information section in the Edit Telephone Numbering page?

You cannot update the country information for a country if the regional information such as area codes, mobile prefixes, and phone formats are defined for the country. For example, if area codes, mobile prefixes, and phone formats are defined for the United States, then you cannot edit the country information.
Data Import Objects: Explained

Import objects are business entities that can be imported into the registry, for example, competitors, partners or resource teams. When you create a data import batch you should choose which business entity, or object, you are importing from the batch into the registry. For example, if you are responsible for resource management, you might want to import objects such as employee resource and resource team.

The import process flow will change according to which object you have selected. There are two import process flows for the following sets of objects:

2. Employee resource, resource team, partner, and partner contact.

Customer, Reference, Competitor, and Custom Party

When you select these objects you will receive the option to check for duplicates within the import batch before the import, and the option to check for duplicates between the import batch and the registry before import. You will also be able to choose to preview data before it is imported, specify if addresses will be cleansed before import, and set how many errors you will allow before the import is terminated.

Employee Resource, Resource Team, Partner, and Partner Contact

If you choose to import these objects you will not be able to deduplicate the batch or registry data. However, you will be able to choose to preview data before it is imported, specify if addresses will be cleansed before import, and set how many errors you will allow before the import is terminated.

Using Data Quality Services During Data Import: Explained

Data Quality Services help you ensure the quality of data being imported into the registry. You can select how to use the services before importing data into the registry.
The data quality services are:

- **Batch Deduplication**: Batch deduplication allows you to define deduplication within the data being loaded.
- **Registry Deduplication**: Registry deduplication allows you to define deduplication of the data being loaded against the records that already exist in the registry.
- **Import to Registry options**: Import to Registry options allow you to define the import process mode, data cleansing, and geography validation.

You can configure a data import or file import process to invoke these services before importing data into the database.

**Defining Data Quality Services for File-Based Import**

The file-based import process allows you to define process criteria, file mapping, and import schedule for importing external files containing business objects into staging tables. You can define Registry deduplication options in the file-based import process. You cannot define the other Data Quality Services, such as Batch Deduplication and Import to Registry options, for a file-based import activity during the file-based import process. However, an import activity can be paused and sent for administrator review after preprocessing and before importing the data. The administrator such as a data steward can then review the import process and configure data quality services.

Import activities are paused and sent for administrator review if the HZ_IMP_PAUSE_FILE_IMPORT profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page. The administrator can then review and edit import activity details and options. Based on the review, the administrator can decide to approve and schedule the import batch or reject the import batch.

**Defining Data Quality Services for Data Import**

You can configure data quality services for data import in the Define Import pages. For both the deduplication methods, you can select a match configuration to identify duplicates and also specify the action to perform on the duplicate records. In the Define Import: Import to Registry page, you can specify whether to run the import process in preview mode, whether to cleanse data before import, and whether to perform geography validation of data before import.

You can also configure geography validation for data import at the site level using the HZ_IMP_DEFAULT_GEO_VALID_ADDRESS profile option in the Manage Import Profile Options page. The addresses are validated if the profile option is set to Yes in the Manage Import Profile Options page. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported.

**Defining the Import Process for Customers and Consumers: Points to Consider**

In Oracle Sales Cloud, you can use File-Based Import to import data directly from a CSV or XML file. In addition, you can use the Data Import dashboard to
monitor the import job and view reports and statistics. You can choose to pause a File-Import batch, and use Data Import work area to configure data quality services, monitor import jobs, view errors, and view batch statistics. Import activities are paused if the HZ_IMP_PAUSE_FILE_IMPORT profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page.

---

**Note**

In on-premise deployment, you can use Data Import to import data from interface tables once the tables are populated by an external process. However, you cannot use Data Import in a Cloud deployment, because you cannot populate interface tables using an external process in a Cloud deployment.

---

Before importing data into the registry, you need to decide if you want to use the data quality services and if so, how you want to configure the data quality services.

The data quality services are:

- Batch deduplication
- Registry deduplication
- Import to registry options

**Batch Deduplication**

Within the batch deduplication page you can decide if you want to identify and resolve duplicates within the batch that you are importing from the interface tables. If you want to check for duplicates you need to choose what match configuration rule you want to use to identify duplicates for each entity. Then you need to specify what action will be taken on the persons, organizations, and address duplicates found within the batch. Your specified actions will be performed on the batch before the data is imported into the registry.

**Registry Deduplication**

Similar to batch deduplication, registry deduplication identifies duplicates between the data in the batch and the data in the registry before the data is imported into the registry. If you want to check for duplicates you need to choose what match configuration rule you want to use to identify duplicates for each entity. Then you need to specify what action will be taken on the persons, organizations, and address duplicates found in the registry deduplication check. Your specified actions will be performed when you import the batch into the registry.

**Import to Registry Options**

When defining an import process you can decide whether to run the import process in preview mode, or you can choose to load the data directly into the registry without previewing the data. You can also choose to cleanse addresses prior to import, validate addresses in an import batch against geography data, and define an error limit for the batch.
**Import Process Mode**

You can choose to run the import batch in preview mode, or you can skip the preview and load the data directly into the registry.

If you select to run the batch in preview mode you will be able to review information about the level of duplicates or incorrect addresses in the batch data before the data is actually imported. You will also be able to preview how many records will be created and how many records will be updated for each entity. You can then continue to import the batch, or you can amend the match configuration rules and actions to be taken on the identified duplicates and then rerun the batch to review the data again.

If you do not want to review the batch data before it is imported into the registry, then you can choose to skip the preview and allow the data to be loaded into the registry as soon as preprocessing is complete. You may prefer not to preview the batch data if the data source is frequently used.

---

**Note**

The Define Import: Import to Registry page is the only place that you can specify if you want to run the batch in preview mode. Once the option to skip the preview mode is selected, and you submit the batch for processing, you will not be able to review the batch data before it is imported.

---

**Cleanse Addresses**

You can choose to validate the addresses in the interface tables before importing them into the registry. The addresses are validated using an integrated third party service that verifies addresses and corrects them if they are incorrect.

**Geography Validation**

You can choose to validate the addresses in an import batch against geography data before importing them into registry. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported. However, Geography Name Referencing is run on all imported addresses regardless of this profile option setting.

**Error Handling Limit**

You can define how many process errors can be generated by the import batch process before the process terminates automatically. Error reports are generated by the application for you to review.

---

**Defining the Import Process for Customers and Consumers: Worked Example**

This example demonstrates how to create an import batch, and how to define the import process data quality services you want to use on the data prior to completing the import.

---

**Note**

Once an import is completed, the data is loaded into the registry.

Create an import batch containing customer and consumer objects, load the import data into the interface tables, and configure the data import process so
that you can view the batch in preview mode to check that all duplicate data are removed.

Creating an import batch

1. On the Data Import Batches Overview page, click on the Create Data Import Batch task located in the task pane.

2. On the Create Data Import Batch page, complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Name</td>
<td>Customer Import Batch</td>
</tr>
<tr>
<td>Source System</td>
<td>Comma separated values</td>
</tr>
<tr>
<td>Object</td>
<td>Customer and Consumer</td>
</tr>
<tr>
<td>Estimated Number of Records</td>
<td>300</td>
</tr>
<tr>
<td>Batch Description</td>
<td>Import of customer data</td>
</tr>
</tbody>
</table>

3. Click Save and Close

4. After creating the import batch, load your data into the interface tables using a standard ETL tool.

Defining the Data Import Process: Batch Deduplication

You want to check the batch for address, organization, and person duplicates; you also want to remove all of these duplicates from the batch.

1. On the Data Import Batches Overview page, highlight the Batch Name, but do not click on the Batch ID URL.

2. Click Actions and then click Import.

3. On the Define Import: Batch Deduplication page, select the Check for duplicates within the batch before import.

4. Complete the fields in the Select match configuration to determine duplicates within the import batch section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addresses</td>
<td>Batch Location Basic Duplicate Identification</td>
</tr>
<tr>
<td>Organizations</td>
<td>Batch Organization Basic Duplicate Identification</td>
</tr>
<tr>
<td>Persons</td>
<td>Batch Person Basic Duplicate Identification</td>
</tr>
</tbody>
</table>
5. Complete the fields in the **Override Default Actions** section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select Action for Persons and Organizations</td>
<td>Remove all duplicates</td>
</tr>
<tr>
<td>Select Action for Addresses</td>
<td>Remove all duplicates</td>
</tr>
</tbody>
</table>

6. Click *Next*.

**Defining the Data Import Process: Registry Deduplication**

You want to check for address, organization, and person duplicates between the import batch and the registry; you also do not want to import duplicate records.

1. On the Define Import: Registry Deduplication page, select the **Check for duplicates between the import batch and the registry before import**.

2. Complete the fields in the **Select match configuration to determine duplicates within the import batch** section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addresses</td>
<td>Batch Location Basic Duplicate Identification</td>
</tr>
<tr>
<td>Organizations</td>
<td>Batch Organization Basic Duplicate Identification</td>
</tr>
<tr>
<td>Persons</td>
<td>Batch Person Basic Duplicate Identification</td>
</tr>
</tbody>
</table>

3. Complete the fields in the Override Default Actions section, as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select Action for Persons and Organizations</td>
<td>Do not import duplicate records</td>
</tr>
<tr>
<td>Select Action for Addresses</td>
<td>Do not import duplicate records</td>
</tr>
</tbody>
</table>

4. Click *Next*.

**Defining the Data Import Process: Import to Registry**

You want to configure the data import process so that you can view the batch in preview mode, enabling you to review the data after preprocessing. You want
to cleanse addresses before they are imported, and specify an error limit for the batch. You also want to validate all addresses in the import batch against geography data. This validates all incoming addresses as per the geography structure and validation level setup in Manage Geographies task. The addresses that are reported as Error during validation are not imported.

1. On the Define Import: Import to Registry page, select Run the batch in preview mode.
2. Select Cleanse addresses before import.
3. Select Validate address against geography.
4. In the Error Limit field, enter 200.
5. Click Submit.

Defining the Import Process for Resources and Partners: Points to Consider

In Oracle Sales Cloud, you can use File-Based Import to import data directly from a CSV or XML file. In addition, you can use the Data Import dashboard to monitor the import job and view reports and statistics. You can choose to pause a File-Import batch, and use Data Import work area to configure data quality services, monitor import jobs, view errors, and view batch statistics. Import activities are paused if the HZ_IMP_PAUSE_FILE_IMPORT profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page.

Note

In on-premise deployment, you can use Data Import to import data from interface tables once the tables are populated by an external process. However, you cannot use Data Import in a Cloud deployment, because you cannot populate interface tables using an external process in a Cloud deployment.

When defining an import process you can decide whether to run the import process in preview mode, or you can choose to load the data directly into the registry without previewing the data. You can also choose to cleanse addresses prior to import, validate addresses in an import batch against geography data, and define an error limit for the batch.

Import Process Mode

You can choose to run the import batch in preview mode, or you can skip the preview and load the data directly into the registry.

If you select to run the batch in preview mode you will be able to review information about the level of duplicates or incorrect addresses in the batch data before the data is actually imported. You will also be able to preview how many records will be created and how many records will be updated for each entity. You can then continue to import the batch, or you can amend the match configuration rules and actions to be taken on the identified duplicates and then rerun the batch to review the data again.
If you do not want to review the batch data before it is imported into the registry, then you can choose to skip the preview and allow the data to be loaded into the registry as soon as preprocessing is complete. You may prefer not to preview the batch data if the data source is frequently used.

**Note**

The Define Import: Import to Registry page is the only place that you can specify if you want to run the batch in preview mode. Once the option to skip the preview mode is selected, and you submit the batch for processing, you will not be able to review the batch data before it is imported.

**Cleanse Addresses**

You can choose to validate the addresses in the interface tables before importing them into the registry. The addresses are validated using an integrated third party service that verifies addresses and corrects them if they are incorrect.

**Geography Validation**

You can choose to validate the addresses in an import batch against geography data before importing them into registry. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported. However, Geography Name Referencing is run on all imported addresses regardless of this profile option setting.

**Error Handling Limit**

You can define how many process errors can be generated by the import batch process before the process terminates automatically. Error reports are generated by the application for you to review.

### Defining the Import Process for Resources and Partners: Worked Example

This example demonstrates how to create an import batch, and how to invoke the data quality address cleansing service you want to use on the data prior to completing the import.

**Note**

Once an import is completed, the data is loaded into the registry.

Create an import batch containing partner objects, load the import data into the interface tables, and configure the data import process so that you can view the batch in preview mode to check the address cleansing results.

**Creating an import batch**

1. On the Data Import Batches Overview page, click on the Create Data Import Batch task located in the task pane.
2. On the Create Data Import Batch page, complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Name</td>
<td>Customer Import Batch</td>
</tr>
<tr>
<td>Source System</td>
<td>Comma separated values</td>
</tr>
<tr>
<td>Object</td>
<td>Customer and Consumer</td>
</tr>
<tr>
<td>Estimated Number of Records</td>
<td>300</td>
</tr>
<tr>
<td>Batch Description</td>
<td>Import of customer data</td>
</tr>
</tbody>
</table>

3. Click **Save and Close**

4. After creating the import batch, load your data into the interface tables using a standard ETL tool.

### Defining the Data Import Process: Import to Registry

You want to configure the data import process so that you can view the batch in preview mode, enabling you to review the data after preprocessing. You want to cleanse addresses before they are imported, and specify an error limit for the batch. You also want to validate all addresses in the import batch against geography data. This validates all incoming addresses as per the geography structure and validation level setup in Manage Geographies task. The addresses that are reported as Error during validation are not imported.

1. On the Data Import Batches Overview page, highlight the Batch Name, but do not click on the Batch ID URL.
2. Click **Actions** and then click **Import**.
3. On the Define Import: Import to Registry page, select **Run the batch in preview mode**.
4. Select **Cleanse addresses before import**.
5. Select **Validate address against geography**.
6. In the **Error Limit** field, enter 200.
7. Click **Submit**.

### Performing What-If Analysis on Data Import Batches: Worked Example

This example demonstrates how to perform What-If analysis on a data import batch that has been processed and has completed with a status of pre-import completed. The match configuration is redefined and the import process is
resubmitted. The batch deduplication actions are then amended, and the batch import is completed.

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>Do you want to redefine batch deduplication match configuration?</td>
<td>Yes, a different match configuration is selected for the organizations entity.</td>
</tr>
<tr>
<td>Do you want to redefine registry deduplication match configuration?</td>
<td>Yes, a different match configuration is selected for the persons entity.</td>
</tr>
<tr>
<td>What actions do you want to take on Persons, Organizations, and Addresses duplicates?</td>
<td>Within registry deduplication, choose Do not import duplicate records for Persons and Organizations. Within registry deduplication, choose Import duplicate records for Addresses.</td>
</tr>
</tbody>
</table>

**Prerequisites**

1. The data import batch has been created.
2. The data is uploaded into the interface tables.
3. The batch is imported and has completed with a status of pre-import completed.

**Viewing the What-If Analysis**

1. On the Data Import Batches Overview page, click on the batch ID URL.
2. On the Edit Data Import Batch page, review the summary and import process performance information. Click **Import Details** to open the What-If analysis page.
3. On the Import Process Details page, click the **Batch Deduplication** tab. Check that you are satisfied with the batch deduplication results.
4. On the Import Process Details page, click the **Registry Deduplication** tab. Check that you are satisfied with the registry deduplication results.
5. On the Import Process Details page, click the **Address Cleansing** tab. Check that you are satisfied with the address cleansing results.

**Redefining the Match Configuration and Resubmitting the Import Process**

The results of the batch and registry deduplication are not as expected and so the match configurations need to be redefined.

1. On the Import Process Details page, click **Cancel**.
2. On the Data Import Batches Overview page, click on the batch name. Click **Actions** and then click **Import**.
3. On the Define Import: Batch Deduplication page, choose a different match configuration for the organizations entity. Click **Next**.
4. On the Define Import: Registry Deduplication page, choose a different match configuration for the persons entity. Click **Next**.
5. On the Define Import: Import Registry page, click Submit.

**Changing the Action for Duplicates within the What-If Analysis**

You want to view the What-If analysis for the new match configurations that you selected for the batch.

1. On the Data Import Batches Overview page, click on the batch ID URL.
2. On the Edit Data Import Batch page, review the summary and import process performance information. Click Import Details to open the What-If analysis page.
3. The new match configurations have produced satisfactory results, but you would like to change the actions that will be carried out on the duplicates. On the Import Process Details page, click the Registry Deduplication tab.
4. For the Persons and Organizations duplicates, choose **Do not import duplicate records** from the choice list.
5. For the possible duplicates for Addresses, choose **Import duplicate records** from the choice list.
6. Click **Complete Import**.

**FAQs for Define Data Import**

**Can I redefine the data import process for an already imported batch and reimport it?**

Yes. If the data is still available in the interface tables and the batch status is Preimport Completed, Completed with Errors, Error, or Terminated when Error Limit Reached, then you can redefine the data import process and reimport the batch. However, once a batch has been successfully imported then you will not able to reimport the batch, even if the data is present in the interface tables.

**How can I view the errors that occurred during preimport processing?**

You can view any errors that occurred after submitting the batch for import by selecting the batch in the data import batches Overview page, and then click Report.

**What happens if I purge a data import batch?**

You permanently remove all records in the batch from the import interface tables. You should purge batches after the batch has been imported successfully and you are sure that the data in the interface tables is no longer required.
Purging the interface tables improves import performance. To archive imported data, you should copy the data to a set of custom tables.

**Why can't I purge a data import batch?**

Data import batches cannot be purged when they have a status of **Processing**. Purging a batch purges the batch data in the interface tables and so cannot be carried out while the batch is importing the data from the interface tables into the registry.

**Why did I receive a message that the data enrichment process was declined?**

The customer data quality application may be unavailable, or you may not have the necessary licenses for the data quality application.

**Can I try and import a batch again?**

Yes. You can reimport a batch to correct validation errors if the import process status is either **Preimport Completed**, **Completed With Errors**, **Error**, or **Terminated When Error Limit Reached**.
Common Customer Data Management Configuration: Define File-Based Data Import

File-Based Data Import: Highlights

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

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

Oracle Sales Cloud File-Based Data Import Guide

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

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

Oracle Enterprise Repository Assets

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

- Reference files that contain detailed object-specific import information, such as target import objects and attributes, required import fields, validation requirements, lookup sources, and default values
FAQs for Define File-Based Data Import

What determines the list of objects displayed?

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

What happens if I inactivate an Import Activity?

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

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

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

Resource Directory: Explained

The Resource Directory offers detailed information about all the resources within the deploying organization. The Resource Directory also enables you to find and communicate with other resources, and to network and collaborate with them.

Use the Resource Directory to perform the following tasks:

- View and modify your profile
- View your organization and team membership information
- View information related to other organizations and teams
- View the profiles of other resources
- Communicate with other resources

Setting up Resources: Explained

Setting up resources involves identifying a person as a resource and specifying optional profile details as needed. This is an important step because until you identify users as resources, you cannot assign work objects to them.

While identifying a resource is the only mandatory task in resource setup, you may also need to perform some of the following tasks while setting up resources.

- Specify the end date for a resource's engagement with the deploying company
- Assign roles to resources
- Assign resources to organizations
Assign resources to teams

Identifying Resources: Explained

The Identify Resources step in the Manage Resources task is only needed to identify an existing employee, contingent worker, or partner member as a resource. Usually they are identified as resource in the Manage Users task, or in the Partner Center. If you have created partner members or internal users in the system without making them resources, you can identify them as resources in the Identify Resources step. Until you identify employees, contingent workers, and partner members as resources, you cannot assign them work objects.

Note

Resources need not necessarily belong to an organization, nor do they need to have specific roles assigned. However, it is best to always associate resources with an organization either as managers or as members. Similarly resources should also have at least one role as part of their organization membership. When you identify users as resources, all you indicate is that these new resources can now be assigned work within the deploying company.

Resource Skills and Resource Assignment: Explained

Resource skills help you assign resources to organizations and teams which can best utilize a specific set of skills. For example, if a resource is skilled in a specific technology, product, or business domain, you can assign the resource to teams and organizations that need resources possessing such skills. Use skill-based resource assignment to get the best out of the resources available to the deploying company.

Resources: How They Work within a Team

You can include resources from different resource organizations to work together on a work object as members of the same resource team. You can also include entire resource organizations into a resource team. Generally what resources can do is controlled by their resource organization membership and their hierarchy. Resource teams provide a flexible way of bringing resources together without any organizational or hierarchy-based restrictions.

Assigning Resources to Teams

You can assign identified resources to teams and assign them roles within the team. Each resource can have a specific role within a team. Thus, a resource may play different roles in different teams.
FAQs for Define Resource Information

What happens if I delete a resource?

When you delete a resource, the resource is deleted from the Resource Directory and from the organizations and teams with which the resource was associated. It is always best, though, to check a resource’s usage and remove the resource from all teams and organizations before deletion from the resource directory.

What happens if I add a resource to the organization?

When you add a resource to an organization, the resource becomes a member of the organization. This positions the resource within the organization hierarchy. Organization membership information is part of the publicly visible details of a resource profile. This means that a resource’s organization membership and reporting structure are visible to all active resources within the organization. If you assign the entire organization to a resource team, all member resources are automatically assigned to the team. This information also becomes part of the resource’s publicly visible profile.

What’s the difference between a partner resource and an internal resource?

The main difference between an internal resource and a partner resource is the company for whom each works. While the internal resource is an employee or contingent worker of the deploying company, the partner resource is an employee of the partner company.

The methodology used to create resources of these two types is also different. While the partner administrator or channel manager creates a new partner resource through the Oracle Fusion Partner Management applications, internal resources are added using the Manage Users, Hire Employee, or Import Person and Organization task.

Another difference between partner resources and internal resources is that partner resources cannot access the Resource Directory while internal resources can.

Can I create an employee or contingent worker resource?

No. You can only identify existing employees and contingent workers as resources in the Manage Resources task, but you cannot create a new employee or contingent worker in the Manage Resources task.
You can create an employee or contingent worker using Manage Users task, Hire Employee task, or Import Person and Organization task.

**What happens when a resource quits?**

When resources quit, their status is set to inactive. This automatically removes these resources from all organizations and teams in the deploying company, and no new work objects can be assigned to them.

**How can I identify a resource?**

To identify specific employees, contingent workers, or partner contacts as resources, you need to search for them and enable them as resources. Once these new resources have been identified, you can assign them to organizations and teams as needed.

Use the Manage Users setup task to create new employees or contingent worker users. New employees or contingent workers can also be identified as resources similarly.
Define Resource Roles: Explained

Define resource roles to create roles that can be assigned to resources within resource organizations or teams. All resource roles have security privileges associated with them, and when you assign a specific role to resources, they automatically receive access to specific business functions, locations, applications, and data within the company. Resource roles are also required for Oracle Sales Cloud, and form the basis of enterprise role provisioning.

The initial seeded data includes resource roles that correspond to the seeded job, or enterprise, roles. Use the Manage Resource Roles task to create additional resource roles or to modify seeded roles. You can access the Manage Resource Roles page by searching for the task in the Navigator > Setup and Maintenance page.

You can set several flags while defining roles. Use these flags along with the organization hierarchy information to define the reporting hierarchy of resources.

- Use the Manager flag to tag a role as a supervisor role. You must assign a resource organization to a manager resource, who will have visibility into all opportunities assigned to the members of that resource organization.
- Use the Member flag to make a role a subordinate role in the resource hierarchy.
- Tag roles as Administrator or Lead to indicate the nature of the tasks that resources who are assigned these roles perform within the hierarchy.

Additionally, you can use these flags along with the resource organization hierarchy information to maintain manager-to-manager relationships within the organization. For example, if Resource A is a manager and heads a sales organization, and Resource B heads his own sales organization, then both Resource A & Resource B will report into Resource C, the Sales Vice President. Similarly, you can create resource organizations and manager resources, and relate managers to higher-level managers, building the resource organization and reporting hierarchies of the company.
Assigning Resource Role Types

Resource role types organize roles into logical groups. This simplifies role assignment and assignment tracking. For example, the Partner resource role type defines a set of partner-specific roles, such as partner sales representative and partner sales manager. Use the Sales resource role type and the Marketing resource role type to categorize the appropriate sales and marketing roles for internal employees or contingent worker resources.

Provisioning Security Roles: Explained

Security role provisioning is the process of automating the provisioning and de-provisioning of security roles based on resource role assignment to resources. Once security roles are provisioned to resources, they can access the tasks and data enabled for the security role.

Difference between Resource Roles and Security Roles

Resource roles indicate who a person is to the deploying company. As such, resource roles are used for filtering resources and for generating reporting hierarchies in addition to being used to define security policies. A key difference between a security role and resource role is that a resource role may be assigned to a resource without a user account, while a security role can only be provisioned to a resource who has a user account. So while in some cases the resource role may be defined at the same granularity as the security role and used to automate security role provisioning, the resource role concept remains separate from security roles.

Job-to-Role Mapping: Explained

In the Manage Resource Roles task, you can establish job mapping for a resource role. Job-to-resource-role mapping enables you to associate HCM jobs with specific resource roles. This mapping simplifies the task of assigning resource roles to new employees or contingent workers, resulting in time and costs efficiency.

For example, suppose a new employee joins the IT department as a data quality manager. If the new employee’s job is already mapped to a resource role like Data Steward Manager, the resource role is automatically assigned when the employee is identified as a resource in the system. This enables you to place new employees faster in organizational and reporting hierarchies. If security roles are also associated with the resource role, then the new employee’s access privileges are also granted automatically.

Resource Role Assignment: Explained

Resource role assignment is the task of assigning roles to active resources within the deploying company. These roles are previously set up, and have associated
security privileges. Thus, when you assign a specific role to resources, they automatically receive access to specific business functions within the company.

You can assign different roles to the same resource in different organizations. Thus, the same resource can have access to different business functions depending on the security roles associated with the resource roles assigned to them.

Resource Team Membership and Role Assignment Components: How They Work Together

Resources who are team members can be assigned different roles within the team. These roles do not necessarily need to reflect the roles these resources may play in resource organizations. Depending on the requirements of the tasks assigned to the team, resources may be assigned resource roles, and can accordingly access data related to the tasks in which the team is engaged.

Resource Team Membership

A resource can belong to multiple teams depending on the needs of these teams and the skills that the resource offers. This does not affect the resource's membership with organizations within the deploying company.

Role Assignment

Resources have specific roles to play in every team to which they are assigned. Each of these roles can be different. Also, these roles can be different from the roles assigned to the same resources in resource organizations. Thus, a resource can be a manager in one team, and a member in another simultaneously.

FAQs for Define Resource Role Information

Can I add more than one resource role to a resource?

Yes. While a resource can have only one resource role within a resource organization, a resource can have multiple roles in different organizations across the deploying company.

Can I assign more than one resource role to a resource organization member?

No. One resource can only have one role within an organization. However, a resource can belong to more than one team, and can have multiple roles in each.
Common Customer Data Management Configuration: Define Resource Organization Information

Resource Organizations and Organization Usage: Explained

You can assign organization usage information to resource organizations to classify them based on how they can be used. For instance, resource organizations engaged in sales activities can be assigned the Sales Organization usage. This enables you to sort organizations based on their usage, simplifying your task of working with them.

Primary Resource Organization by Usage Explained

A resource can belong to multiple organizations, and can be assigned different resource roles for each organization. However, a resource can be assigned to one resource organization that becomes the primary organization for the resource as far as a specific usage is concerned.

A resource organization can be said to be a primary resource organization by usage for a resource if the following criteria are met.

- The resource must be a member of the concerned organization.
- The resource organization must be classified as an organization with the specific usage.

For example, a resource organization can be listed as a primary resource organization for marketing for a particular resource if the resource organization is classified as a marketing organization by usage, and has the concerned resource as a member.

Resource Organization Hierarchy Versioning: Explained

As organizations evolve, you may need to make changes to the existing organization hierarchy. Create organization hierarchies to capture these changes without impacting active hierarchies.
Depending on the urgency and nature of the changes within the deploying company, organization hierarchy changes can either be immediate or planned.

**Implementing Immediate Resource Organization Hierarchy Changes**

In case of immediate changes in the organization hierarchy, either make changes directly to the hierarchy or create a new version of the existing hierarchy and set it to become active when the new organization structure needs takes effect.

**Note**

Changes made to existing hierarchies are saved automatically and updated immediately.

**Implementing Planned Resource Organization Hierarchy Changes**

Create a new version of the active hierarchy and specify the date on which the new version needs to become active. Once the new version is saved, you can make and save the changes needed. Ensure that you have made all the changes needed to the new version before the date on which the new version needs to become active.

**Organization Memberships and Role Assignments: Explained**

When you assign a resource to an organization, you should also specify the role the resource needs to play in the organization. This ensures clarity on the kind of tasks that can be assigned to a resource, apart from defining where the resource is situated within the organization’s hierarchy.

**Note**

There can be only one manager in an organization. However, a resource can be a member of multiple organizations if needed.

**Create an Organization with Sales Usage: Worked Example**

This example demonstrates how to create a resource organization with sales usage.

you have two choices while creating resource organizations with sales usage:

- Create an organization from an existing department
- Create a new organization

**Accessing the Managing Sales and Marketing Organizations Screen**

These tasks can be accomplished using the Manage Sales and Marketing Organizations screen in the Setup and Maintenance section. To navigate to this screen, perform the following tasks.

1. Click the Navigator link at the top-left section of the screen.
2. Click the **Setup and Maintenance** link in the Navigator. This displays the Setup and Maintenance screen.

3. If the **Manage Sales and Marketing Organization** task is listed in the **Assigned Implementation tasks**, click the **Go To Task** button against it to access the **Manage Sales and Marketing Organizations** screen.

4. You can also choose to search for the task in the **All Tasks** tab.

### Creating an Organization from an Existing Department

You need to create an organization with sales usage from an existing department if the concerned department needs to be reused as a sales organization.

1. You can either search for the organization whose usage you wish to modify or click the **Create** button, select the **Create from Existing Department** option and specify the name of the organization in the **Create Organization** screen.

2. Specify the **Business Unit** to which the organization needs to belong and add **Sales Organization** under the Organization Usages section of the screen.

### Creating a New Organization with Sales Usage

1. Select the **Create New Organization** option in the **Create Organization** screen and click **Next**.

2. Specify the name and business unit of the new organization and select **Sales Organization** as its usage.

3. Click **Finish** to save the new organization with sales usage.

### Adding a Sales Organization to a Sales Organization Hierarchy: Worked Example

This example describes how you can add a sales organization to a sales organization hierarchy.

As new sales organizations are created within your company, you need to add these organizations to your company's sales organization hierarchy. To do so, perform the tasks detailed below.

**Decide whether to Create a New Version**

1. Navigate to the **View Organization Hierarchies** screen in the Resource Manager.

2. Search for and display the sales organization hierarchy you want to edit.

3. Click the **Actions** drop-down button. If the new organization needs to be added to the sales organization hierarchy with immediate effect, select the **Edit this Hierarchy Version** option. If the new organization needs to be displayed in the sales organization hierarchy only after a specific date, select the **Create Hierarchy Version** option. If you choose the latter option, you need to specify the date from which the new sales organization hierarchy version needs to be active.
**Add a New Organization to the Sales Organization Hierarchy**

1. In the **Edit Organization Hierarchy Version** page, select the node in the sales organization hierarchy where you need to add the new sales organization and click the **Add** button.

2. Search and select the new sales organization in the **Add Tree Node** popup screen. This adds the new organization to the sales organization hierarchy.

3. Click on the new organization in the hierarchy, drag it to the node where it needs to be placed, and release. The organization is now moved to the location you specified within the selected sales organization hierarchy.

4. Click **Save and Close** to save the changes made to the sales organization hierarchy version and exit the screen.

**Assigning a Resource to an Organization: Example**

Once you have identified a resource, you can assign the resource to organizations within the deploying company. This example explains how you can assign resources to organizations.

**Scenario**

You need to assign a resource to an organization. It is important to state here that a resource can belong to multiple organizations at the same time. This means that even if chosen resources already belong to organizations, you can assign them to other organizations as needed.

To assign a resource to an organization, perform the following tasks.

1. Search for the resource using the **Search** controls in the Resource Directory.

2. Display the concerned resource's profile by clicking on the resource's name in the search results.

3. Click the **Organizations** tab in the Resource Profile Details screen. This displays the list of organizations to which the resource belongs.

4. Click the **Create** button to display the **Create Organization Membership** popup screen. Use this screen to search and assign organizations to the concerned resource's profile.

5. Click the **Organization** field and select the organization you need from the drop-down list displayed. You can also enter part of the name of the organization you need and search for it. You can also search for organizations using a range of search options available in the search screen. Click the **OK** button to add the organization you need to the **Create Organization Membership** screen.

6. Click the **Add Row** button in the **Create Organization Membership** screen. This adds a row to the screen, enabling you to specify the role the resource needs to play in the organization.

7. Click the **Role** field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the
role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.

8. The **From Date** field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.

9. The **To Date** is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.

10. Click the **OK** button to save your changes. This assigns the resource to the organization you chose. You can add the resource to as many other organizations as needed.

You can also assign resources to organizations by searching for organizations and adding members to the concerned organization. To assign resources to organizations by searching for organizations, perform the following operations.

1. Click the **Search** field and, from the drop-down list displayed, select **Organizations**.

2. Search for the organization you need and click on the name of the required organization from the search results. This displays the **Organization** screen, where all the details of the selected organization are displayed.

3. Click the **Members** tab to edit the list of members assigned to the organization.

4. Click the **Create** button to assign a new resource to the organization. This displays the **Add Resource Member** popup screen.

5. Search for the resource by entering the resource's name in the **Resource** field. From the search results displayed, select the resource you wish to assign to the organization and click **OK**. This displays the **Add Resource Member** popup screen with the resource's name added.

6. Click the **Add Row button to add a role to the newly-added resource.**

7. Click the **Role** field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.

8. The **From Date** field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.

9. The **To Date** is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.

10. Click the **OK** button to save your changes. This assigns the resource to the organization you chose. You can add as many resources to the organization similarly.

**Note**

1. The date from which a resource is a member of an organization must be on or after the date on which the resource was hired.

2. If a resource was previously a member of the selected resource organization with an end date later than the current date, and if the
FAQs for Define Resource Organization Information

Can I create a new organization?

Yes. You can create new organizations using the Manage Sales and Marketing Organizations option in the Functional Setup Manager or the Oracle Fusion Partner Management user interface. You can also create new resource organizations from the User Management interface. You can also use the interface to manage user accounts and roles, and to create employee and partner-member users.

How can I delete a resource from an organization?

To delete a resource from an organization, you need to delete the concerned resource’s membership from the organization.

To manage the membership of an organization, navigate to the Manage Resource Organization and select the organization concerned. Once the organization’s membership details are listed, you can add, delete or edit them as required.

What happens if I delete a resource from an organization?

When you delete a resource from an organization, you effectively delete the concerned resource’s membership with the organization. This also removes the resource from the organization hierarchy.

What’s a resource organization hierarchy?

A resource organization displays the relationships between the resource organizations of the deploying company. The resource organization hierarchy is visible to all resources within the company.

Can I create a new organization hierarchy?

No. All organizations within your deploying company are part of a large overarching global hierarchy. Therefore, new organization hierarchies cannot be created. You can, however, move existing organizations or attach new
organizations to the existing hierarchy and create a new version of the hierarchy. Once you have created a new version, specify the date from which the new version needs to become active and, on the specified date, your new hierarchy version replaces the older hierarchy.

**Can I copy a resource organization hierarchy?**

No. You can only have one resource organization hierarchy active at a time within a deploying company. If you need to create a hierarchy for an organization, you will need to specify the reporting structure within the concerned organization and then make the organization a part of the existing hierarchy.

**How can I reuse a resource organization in multiple organization hierarchies at the same time?**

There is always one resource organization hierarchy for internal resource organizations and also one partner organization hierarchy for each partner company. You can have one active version and multiple inactive versions of resource organization hierarchies. Resource organizations can be included into any of the active or inactive organization hierarchies.

**Can I create multiple resources as managers within a resource organization?**

No. An organization can have only one manager.

**Can I create a new organization hierarchy version?**

Yes. To create a new hierarchy version, navigate to the hierarchy whose new version you wish to create and click on the **Actions** button. From the drop-down list displayed, select **Create Hierarchy Version**. Make the changes to the hierarchy as required, specify the dates between which the organization hierarchy needs to be active, and click **Save and Close** to save your changes and exit the screen.

**What’s a partner organization?**

A partner organization represents a department, section, division, business unit, or any other internal or external unit of a partner company registered with the deploying company. Partner organizations provide the framework that enables the deploying company to engage with the partner company while performing sales and marketing activities.
You have to be either the channel manager of the deploying company or a designated partner administrator to create a partner organization. To create a partner organization, you should first register the partner company as a partner of the deploying company. Once registration is complete, you should enroll the partner company into a partner program. Once the partner company has been enrolled into a partner program, you can create and manage partner organizations to model the organization hierarchy of the partner company.
Common Customer Data Management Configuration: Define Resource Team Information

Resource Setup and Resource Role Setup: How They Fit Together

Setting up resource roles helps you define the various roles within your organization so that they can be assigned to resources.

How Resource Role Setup Simplifies Resource Setup

Use the Resource Role Setup tasks to define specific roles that best reflect the way responsibilities are distributed within the deploying company. If your resource roles are created before you start setting up your resources, you can easily assign roles to resources, simplifying task assignment. Resource roles that are tagged as manager or member roles also determine how eventually the reporting hierarchy gets built.

Later, when you need to assign resources to organizations, resource roles can help you specify the precise roles that resources need to play within the organizations to which you assign them.

Resource roles also simplify security provisioning for resources. You can define security roles corresponding to resource roles using the role mapping interface. Once such mapping rules are setup, you can provision security roles by assigning the corresponding resource roles to resources.

FAQs for Define Resource Team Information

What's a resource team?

A resource team is a group of resources formed to work on work objects. A resource team may comprise resource organizations, resources, or both.
A resource team cannot be hierarchically structured and is not intended to implement an organization structure. You can also use resource teams as a quick reference to groups of related resources that you can quickly assign work objects to.

**Note**

Members of teams can either be reassigned separately, or entire teams can be assigned to other tasks as required.

**What's a duty role?**

A duty role is a set of privileges associated with an aspect of a resource role, and one or more duty roles roll up into a resource role. Basically, resource roles can be said to comprise a set of duties, and each duty has its own set of privileges. Thus, privileges roll up into duty roles, and duty roles roll up into resource roles.
Managing Accounting Calendars

The accounting calendar is used for sales forecast and territory metrics. You need to set up the calendar to match your fiscal calendar and forecasting periods. If you are loading historical data, then you must set the start date to the first date of your historical data.

**Important**

You cannot change the settings once you start using the application, so consider setting the date to the first date that your company was created. This will allow you to upload historical data later, if necessary.

You can choose the calendar option of your choice: 4-4-5, 4-5-4, 5-4-4, 4 Week, weekly, monthly, quarterly, yearly, or your custom calendar.

**Note**

- This step is required for territory management and forecasting.
- You can change the values until your first calendar period is open.

**Creating the Accounting Calendar**

1. In the Setup and Maintenance area, Overview page, All Tasks tab, search for the topic names containing **Accounting Calendar**.

   Manage Accounting Calendars will show up in the results box.

2. Click the **Go To Task** icon.

3. On the Managing Accounting Calendars page, click the **Create** icon.

4. Fill in the required fields as appropriate for the calendar, and click **Next**.

   Make note of the name utilized because this name will be required in the
step associating the calendar profile option with the calendar. See the topic, Managing the Calendar Profile Option, for more information.

**Note**

When creating the calendar, the first calendar date must be the first date of the period of the oldest historical data on which you will be reporting. For example, if you select January 1, 2010 as your first calendar date, then you would only be able to enter or import historical data associated with this date and later.

5. Ensure that the calendar data is correct, and click **Save**.
Common Customer Data Management Configuration: Manage Bulk Data Export

Bulk Export: Overview

The Bulk Export application provides a mechanism to extract large volumes of data from Oracle Sales Cloud objects. These extracts can be the full set of records for an object or incremental extracts. For example, data extracted for a specific period of time, from the hosted cloud services to an on-premise database that resides behind a user’s fire-wall. The system will create comma separated variable or tab delimited files with the extracted data, which will be available to users as attachments to the batch records that have been executed.

The following figure depicts the process of selecting data for export, scheduling and finally delivering the exported data file.
Bulk Export Process Definition: Explained

This solution provides a mechanism to extract large volumes of data from Oracle Sales Cloud objects, both as extracts of a full set of records for an object as well as incremental extracts. The system will create comma or tab delimited files with the extracted data which will be available to users as attachments to the batch records that have been executed.

In order to create the extracts, two steps must be completed. First, mapping files for the full and incremental extract processes must be defined in the Oracle Sales Cloud. These maps will specify which columns and filters will be applied to each export process for each export object. For the incremental extracts, filters can be created that leverage time stamps to determine which rows will be queried out of the system. All mapping files will be saved in the system and reused for each extract.

Next, the hourly and weekly data export processes are scheduled in the export tool. For any required incremental and scheduled export, the export task should either exist or created through the UI. Oracle Web Services would only be used to schedule the export and start it. After each export process executes and completes, a comma or tab delimited data file is created and stored as an
attachment. The formatted file can be downloaded by using the `getAttachment()` web service or by using the interactive UI in the export tool.

There are no transactional steps for this process in the Oracle Sales Cloud, there are only prerequisite setup steps. Once these steps are complete the process should run automatically. The prerequisite steps in Fusion are to create an export map and export job schedule for each object to be extracted (this only needs to be done once).

The Bulk Export Process Definition is made up of the Export Map and the processing schedule. See the steps below.

**Select the Export Object**

The export object is the data base object where the data resides, and is made up of attributes. If you need to export data from a custom table, you must register the object as an export object. This is accomplished from the Manage Export Process UI, Manage Export Objects action. All the delivered tables and their attributes are available for export.

**Select Attributes**

The export object is made up of attributes. These attribute may be selected for export or not included. You can edit the header text of the attribute to make its meaning more clear to other users of this process.
Filter Criteria

Each attribute may have limits or conditions enforced. Various operators are available for selecting the data to precisely select the data required for the export. You can save the filter criteria and then modify the criteria and save it under a new name. You can then change the filter by coming here to select an alternate filter name. Because the filters are related to the export object, if you reuse a map and change the filter, you are changing it for any Export Process Definition that uses that map. The attributes you use for the map have no bearing on what is available in the filter. All fields from the VO are available for use in the filter. For example, you can filter by TYPE but not show TYPE in the output.

Schedule the Export Process

Once defined, the export process is scheduled. You can run the process immediately or at the time and date of your choosing. If you decide to schedule the job at a later date you can also choose to set up a recurring schedule of extracts.

Activate the Job

By clicking on the Activate button, you make the job available to be run. It does not start an export process.

Bulk Export Process Components: How They Work Together

In the two step process used by Bulk Export, the first is the mapping of files for the full and incremental extract processes. The second step is the scheduling of the export. You create a process definition that includes both of these steps.

The process definition has three components that together make exporting data easier by leveraging the export maps that you have already built. The process name, the export process ID and the export map ID all serve to identify the specific process definition as well as leverage your work with reusable export maps.
Process Name

A user-supplied, natural language way to refer to the Export Process Definition. This enables you to refer to the export process definition easily rather than using the machine generated ID. For example, use Customer or some other meaningful name as the export process name instead of the export process ID 100000019897192.

Export Process ID

A unique, system generated identifier for the export process definition that ties together the export map, with its export objects and filters, and the defined export schedule.

Export Map ID

A unique identifier for the export map itself. You can name the export map or leave the field blank for a system generated map name to be entered. You can reuse the export map in different process definitions. For example, you could create a process definition to export all the data from the Customer export object. You could then reuse that export map and apply a new filter on the data to create an incremental export, such as data accrued since the last export date.
FAQs for Manage Bulk Data Export

How can I tell which objects to select?

Review the requirements for the data to be exported and determine the source view object that holds the attributes you want.

How can I create a subset of data for export?

Full sets of data are not always required for export. To create a subset of data, use filter criteria to determine the time frame or scope of data, based on values of the attributes. For example, to find activities for a certain period, use a project start date from 1/1/11 through 3/31/11, navigate to the Export Objects Detail Sub Page and click the filter icon. Fill in the filter criteria dialog for the project start dates to select the data to be exported. You run the export by navigating to the Setup and Maintenance menu, selecting Manage Task Lists and Tasks. Then, search for Schedule Export Processes and click the Go to Task icon on the line for this task.

How can I see my exported data?

You can look on the Schedule Export Processes, Overview page to see the History subpage. The column Exported Data File shows a hyperlink to your output file. This file will be a comma separated variable or a tab delimited file. Click that link to open the file and see the exported data.

How can I use my own View Object for export?

You can use your own defined view objects as a source for Bulk Export. To register your view objects for export, select Setup and Maintenance from the Tools menu and search for the Manage Export Objects task. Click the Go to Task icon and on the Manage Export Objects page click the Create icon to add your View Object, making it available for use.

What happens if I change the sequence number or header text?

Changing the sequence number changes the order of the attributes in the exported data file. Changing the header text enables you to give a more intuitive meaning to the attribute and the associated data.
What happens if I need data from more than one view object?

Select as many view objects as required to be export objects for the export process. Choose the individual attributes required from each export object.
Enriching Sales Cloud Data Using Oracle Social Data and Insight Cloud Service

Enriching Account and Account Contact Data: Explained

Data enrichment improves the quality of your existing account or account contact data, and also enriches that data with additional information. You use the Oracle Social Data and Insight Cloud service to enrich your account and contact data in Oracle Sales Cloud.

In order to enrich account data, you must:

1. Export the data you want to enrich from Oracle Sales Cloud to a CSV file.
2. Enrich the exported data using the Oracle Social Data and Insight Cloud service.
3. Download the enriched data file from Oracle Social Data and Insight Cloud service.
4. Import the enriched data file into Oracle Sales Cloud.

Exporting Data from Oracle Sales Cloud

You can export your account or account contact data from Oracle Sales Cloud using the Schedule Export Processes task. The export process provides two predefined objects, one each for exporting account and account contact data. While exporting data, you must choose the Organization for Data Cloud Enrichment object to export account data and the Person for Data Cloud Enrichment object to export account contact data. You can also configure the export objects to select the object attributes that you want to enrich.

Enriching Data in the Oracle Social Data and Insight Cloud Service

You can use the Enrich Data functionality of the Oracle Social Data and Insight Cloud service to enrich account or account contact data. To enrich data, select the CSV file exported from Oracle Sales Cloud in the Enrich Data Records window of the Oracle Social Data and Insight Cloud service. You must select the CRM
template during enrichment, so that the enriched file can be downloaded in Oracle Sales Cloud. After enrichment is complete, a CSV file with the enriched data is available for download from the Jobs List.

**Importing Enriched Data into Oracle Sales Cloud**

To import the account or account contact data, import the CSV file you downloaded from Oracle Social Data and Insight Cloud service into Oracle Sales Cloud using the file-based import functionality. You create an import activity, enter the appropriate import details, provide the file mapping, and schedule the import to import the data from the CSV file into staging tables. Import processing is subject to the settings defined for the import object, mapping, and import activity.

Import objects represent the application and attribute information for business objects that can be imported using external source files. While specifying the import object, you choose the Account to import account data and the Contact to import account contact data.

Import mapping enables you to predefine a mapping between the columns provided in a source file and the attributes pertaining to the objects being imported. Oracle Sales Cloud provides seeded mappings to import account data or account contact data from Oracle Social Data and Insight Cloud service. You must select Seeded Data to view the available seeded mappings. While specifying the import mapping, you choose Data Cloud Account Import to import account data and Data Cloud Contact Import to import account contact data. Oracle Sales Cloud would then automatically map the data in the CSV file to object attributes.

**Importing New Records into Oracle Sales Cloud from Oracle Social Data and Insight Cloud Service: Explained**

Oracle Social Data and Insight Cloud service lets you directly import enriched records to your sales cloud instance. You can import accounts or account contacts from Oracle Social Data and Insight Cloud service into your sales cloud instance.

In order to import data from Oracle Social Data and Insight Cloud service to Oracle Sales Cloud, you must:

1. Export the accounts or account contacts from Oracle Social Data and Insight Cloud service.
2. Import the CSV file into Oracle Sales Cloud.

Before importing account contacts, you must ensure that the accounts related to the contacts exist in Oracle Sales Cloud.

**Exporting Accounts or Account Contacts from Oracle Social Data and Insight Cloud Service**

Before you can export account or account contact data, you must search for the data in the Oracle Social Data and Insight Cloud service, and select the data you
want to export. You can search for accounts and account contacts from the Search window. You can also export all available accounts or account contacts.

After searching for the records you want to export, click Export to export the data. You must select a template to specify the structure of the exported CSV file. You must select **Use CRM Template** so that the exported file can be imported in Oracle Sales Cloud.

**Importing Accounts or Account Contacts into Oracle Sales Cloud**

To import the account or account contact data, import the CSV file you downloaded from Oracle Social Data and Insight Cloud service into Oracle Sales Cloud using the file-based import functionality. You create an import activity, enter the appropriate import details, provide the file mapping, and schedule the import to import the data from the CSV file into staging tables. Import processing is subject to the settings defined for the import object, mapping, and import activity.

Import objects represent the application and attribute information for business objects that can be imported using external source files. While specifying the import object, you choose **Account** to import account data and the **Contact** to import account contact data.

Import mapping enables you to predefine a mapping between the columns provided in a source file and the attributes pertaining to the objects being imported. Oracle Sales Cloud provides seeded mappings to import account data or account contact data from Oracle Social Data and Insight Cloud service. You must select Seeded Data to view the available seeded mappings. While specifying the import mapping, you choose **Data Cloud Account Import** to import account data and **Data Cloud Contact Import** to import account contact data. Oracle Sales Cloud would then automatically map the data in the CSV file to object attributes.
Define Data Quality

Define Data Quality: Explained

Defining data quality involves setting up two functionalities, matching and address cleansing, by performing related regular and manual setup and maintenance tasks. Manual setup and maintenance tasks are performed on the related third-party data quality (DQ) engine, Informatica Identity Resolution (IIR).

The customer data quality matching and address cleansing services are designed to cater to all applications that use the registry.

Matching is the identification of potential duplicates for organizations, persons, and locations. Potential duplicate records are identified and displayed in the data quality functionality consuming applications during a variety of activities such as duplicate prevention in real-time when a customer record is being created or duplicate identification in batch mode for existing records.

Address cleansing is the process of correction and validation of address data, based on postal requirements. For example, if a user enters valid values for postal code, city, and country, the data quality functionality may return a value for state. Likewise, if a user enters valid values for city, state, and country, the data quality functionality automatically supplies a postal code value.

This figure illustrates the data quality management architecture. The data quality services receive matching or cleansing requests, such as duplicate prevention or address validation, from a variety of consuming applications, and then internally invoke the appropriate third party services based either on a default configuration or a specific configuration passed to the services.
Matching and address cleansing are independent functionalities and are set up separately using the following steps:

- Setting up the third-party data quality (DQ) engine, Informatica Identity Resolution (IIR)

**Note**

You can find the IIR vendor documentation in `<PROV.TOP>/InformaticaIR/pdf` directory within the IIR Installation.

- Setting up the Fusion DQ Connector to IIR, by performing data quality related regular setup and maintenance tasks, such as defining and managing lookup choices and configurations used in data matching, cleansing, and matching index synchronization
- Connecting applications to the third-party data quality (DQ) engine, Informatica Identity Resolution (IIR)

**Managing Lookups**

Review and define lookup values that provide choices for data matching and cleansing, such as address fields, quality error messages, and match tolerances.

**Managing Data Quality Configurations**

Perform the configurations required to enable data quality processes such as data quality matching and cleansing operations.

You can enable data quality processes such as matching and address cleansing for real-time or batch matching and cleansing, and as part of the data import process.

Real-time matching can prevent individual entry of duplicate entities, such as organization, person, or location, into the registry. Batch matching leads to identification of duplicates entries in the registry.

As part of defining and managing data quality configurations, you search, review, and edit seeded matching and cleansing configurations for real-time and batch matching and cleansing of entities.
Managing Server Configurations

Define, review, and update the data quality server configurations to integrate with the embedded data quality engine. Next, associate the server name with the matching and cleansing configurations to perform duplicate prevention, batch duplicate identification, and real-time and batch data cleansing.

Managing Matching Index Synchronization

Review and update matching index synchronization options, to synchronize registry data with the data quality engine repository, such as system and identity tables. View the latest synchronization results, evaluate errors, and reset for continued processing after resolving error conditions.

Manage Server Configurations

Server Configurations: Explained

Server configurations are predefined configurations for third-party data quality servers. You can search, review, and edit server configurations. Although you can edit the configuration name, server address, server port, and values of some of the configuration parameters, you cannot add or delete a server configuration parameter.

There are four predefined server configurations:

- Real-Time and Batch Basic Match Server
- Real-Time Cleanse Server
- Batch Cleanse Server
- Advanced Batch Match Server

Real-Time and Batch Basic Match Server

Used for basic real-time duplicate prevention and batch data matching and duplicate identification. This configuration requires you to synchronize the registry data with the third-party data quality engine repository periodically to ensure appropriate matching indexes are updated continuously.

Real-Time Cleanse Server

Used for real-time data cleansing operations that happen at touch point applications, and so do not need synchronization.

Batch Cleanse Server

Used for batch cleansing operations during which records in a batch are sent to the data quality server for cleansing one at a time, and the data quality engine returns the cleansed record to the registry likewise, one at a time, thus completing the loop.
Advanced Batch Match Server

Used for advanced batch matching that is done to find duplicates amongst a group of records before uploading them to the registry. For example, an acquiring company may do an advanced batch match of the legacy data of the acquired company before uploading the data to the registry. In such a case, data is uploaded to the repository of the data quality engine directly for duplication identification and after the duplication identification processing it is imported to the registry.

Oracle Sales Cloud Data Quality Connector Setup: Procedures

Once you complete the Informatica Identity Resolution (IIR) matching and cleansing setups, you must ensure that Oracle Sales Cloud is connected to the data quality engine.

Connecting Oracle Sales Cloud to Data Quality Engine

To connect Oracle Sales Cloud to the data quality engine:

1. Map Oracle Sales Cloud to IIR Matching server.
   a. Log in to the Functional Setup Manager application as an administrator user.
   b. Under Assigned Implementation Tasks, search for Manage Server Configuration setup tasks.
   c. Select Go to Task. This opens the task-flow page.
   d. Click Search to find all existing configurations.
   e. In the Search Results, select Realtime and BatchBasic Server.
   f. Select Actions - Edit.
   g. Enter the Server Address and Server Port of the IIR Matching Server. The default port is 1666. To know the current port, use `grep SSA_SEPORT=$<PROV.TOP>/InformaticaIR/env/iss`.  
   h. Select Enable Matching.
   i. (Optional) Select Enable High Availability.
   j. Click Save and Close.

2. Map Oracle Sales Cloud to IIR Cleansing server.
   a. Log in to the Functional Setup Manager Application as an administrator user.
   b. Under Assigned Implementation Tasks, search for Manage Server Configuration setup tasks.
   c. Select Go to Task. This opens the task-flow page.
   d. Click Search to find all existing configurations.
   e. In the Search Results, select Realtime Cleanse Server.
f. Select **Actions - Edit**.

**Note**
Repeat the steps to select Batch Cleanse Server from the search results and edit it.

g. Enter the **Server Address** and **Server Port** of the IIR Cleansing server. The default port is 1666. To know the current port, use `grep SSA_SEPORT=\$<PROV.TOP>/InformaticaIR/env/iss`.

h. Select **Enable Cleansing**.

i. (Optional) Select **Enable High Availability**.

j. Click **Save and Close**.

**Note**
By default, the IIR server instance is the same for both matching and cleansing, but you can configure separate instances if required.

---

**Manage Matching Configurations**

**Matching Configurations: Explained**

Matching configurations are predefined configurations used for real-time and batch matching of party entities with the intention of preventing and identifying duplicate entries.

Real-time and batch matching are available for the following entities:
- Organization
- Person
- Location

There are six predefined matching configurations, three each for real-time and batch matching.

These configurations include both customer data quality parameters and the parameters of the embedded data quality engine, Informatica Identity Resolution (IIR). While some of the parameters, such as **MatchTolerance**, are single valued, others, such as **Manage Search Definition**, are multi-valued.

You can search, review, and edit these seeded matching configurations.

**Note**
Although you cannot add or delete a matching configuration parameter, you can change the configuration name and update the parameter values to suit your requirements. For more information, see: Matching Configuration Parameters.

**Real-time Matching Configurations**

The following are the predefined real-time matching configurations:
- Real Time Location Duplicate Prevention
- Real Time Organization Duplicate Prevention
• Real Time Person Duplicate Prevention

Real-time matching prevents individual entry of duplicate organization, person, or location information. Real-time matching finds all possible duplicate records that may exist for an entered record, and assigns a match score to each potential duplicate identified. Based on the match score returned by the application and the threshold settings in the configuration, the user can either select an existing duplicate record or continue to create a new record.

Batch Matching Configurations

The following are the predefined batch matching configurations:

• Batch Location Basic Duplicate Identification
• Batch Organization Basic Duplicate Identification
• Batch Person Basic Duplicate Identification

Batch matching leads to the identification of duplicate entries between existing customer data and new data sets, such as import batches, or within sets of data to resolve them by merging or linking. While real-time matching provides duplicate prevention for only one record and returns potential matches for the record being entered, the batch matching service takes a set of records of the same type, such as person, organizations, or locations and identifies all possible matches within these sets of records. The duplicate data is presented in sets of possible matches with a score associated with each individual record in the set.

IIR Matching Process

Based on the matching configurations provided by the data quality functionality, the embedded data quality engine, Informatica Identity Resolution (IIR), returns the final set of matches by performing a two-step process comprising candidate selection and scoring.

These steps are described below:

1. Candidate Selection: This step requires that the matched-against records are stored in the IIR repository and indexed. IIR uses search definitions to search and shortlist candidate records from among the repository records.

2. Scoring: In this step, IIR does a field by field level comparison of the candidate record against the corresponding fields in the record being matched. The matching fields and the scoring are also defined in the search definition.

Defining Indexes and Index Synchronization

Matching using the configurations provided by the data quality functionality requires both defining indexes, or initial indexing, and index synchronization of the Informatica Identity Resolution (IIR) repository with the existing customer data.

A prerequisite to defining indexes or initial indexing is the creation of a new system on IIR using the system definition file FusionDQRealtime.sdf, shipped with the application, which uses US population and contains information about IIR match rules and indexes.

Before defining indexes, review the index information in the out of the box system definition file FusionDQRealtime.sdf to verify that it is up to date and make any modifications required to suit your specific business needs.
Staging Tables

IIR staging tables, also called identity tables (IDTs), are used to achieve the following:

- Stage Oracle Sales Cloud data.
- Partition the data based on customer data entities and their attributes.

Identity tables (IDTs) are also used for periodic index synchronization of the third party data quality engine repository with the customer data in the application to account for the continual updates made to the registry.

Note

IIR staging tables definitions are stored in the `FusionDQRealtime.sdf` file shipped with the application.

The following table describes the different staging or identity (IDT) tables.

<table>
<thead>
<tr>
<th>Entity</th>
<th>IDT Name</th>
<th>Data Content</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>ORG_PRIMARY_IDT</td>
<td>Profile information and primary attributes, such as primary address and primary phone, of an organization.</td>
<td>The IDT includes one row for each organization.</td>
</tr>
<tr>
<td>Organization</td>
<td>ORG_ADDRESS_IDT</td>
<td>Multiple addresses and primary attributes, such as primary phone, of an organization.</td>
<td>The IDT does not include organizations that have no addresses. For organizations that have addresses, the IDT includes one row for each active address.</td>
</tr>
<tr>
<td>Person</td>
<td>PER_PRIMARY_IDT</td>
<td>Profile information and primary attributes, such as primary address and primary phone, of a person.</td>
<td>The IDT includes one row for each person.</td>
</tr>
<tr>
<td>Person</td>
<td>PER_ADDRESS_IDT</td>
<td>Multiple addresses and primary attributes, such as primary phone, of a person.</td>
<td>The IDT does not include persons that have no addresses. For persons that have addresses, the IDT includes one row for each active address.</td>
</tr>
<tr>
<td>Person</td>
<td>PER_PHONE_IDT</td>
<td>Multiple Phones and Primary attributes (primary phone) of a person, with Key created on &quot;Concatenated Address Lines&quot;.</td>
<td>The IDT does not include persons that have no phones. For persons that have phones, the IDT includes one row for each active phone.</td>
</tr>
<tr>
<td>Entity</td>
<td>IDT Name</td>
<td>Data Content</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Location</td>
<td>LOCATION_IDT</td>
<td>Location Information</td>
<td>The IDT includes one row for each active location.</td>
</tr>
</tbody>
</table>

**Search Definitions**

IIR uses search definitions to search and shortlist candidate records from among the repository records.

The following table describes the different search definitions:

<table>
<thead>
<tr>
<th>Entity</th>
<th>Search Definition</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>org-dup</td>
<td>Uses Organization Name as key or acquisition attribute and scores on other fields, using weights. Organization Name has the highest weight (about 67%), although other attributes contribute, as well.</td>
</tr>
<tr>
<td>Organization</td>
<td>org-address-dup (multi-search definition)</td>
<td>Leads to two searches, one against Organization Name (key 1) and another against Address Line 1 (key 2). So matching on any of these yields a match.</td>
</tr>
<tr>
<td>Person</td>
<td>person-dup</td>
<td>Uses Person Name as key or acquisition attribute and scores on other fields, using weights. Organization Name has the highest weight (about 67%), although other attributes contribute, as well.</td>
</tr>
<tr>
<td>Person</td>
<td>person-address-dup (multi-search definition)</td>
<td>Leads to two searches, one against Person Name (key 1) and another against Address Line 1 (key 2) and combines results. So matching on any of these yields a match.</td>
</tr>
<tr>
<td>Person</td>
<td>person-phone-dup (multi-search definition)</td>
<td>Leads to two searches, one against Person Name (key 1) and another against Phone (key 2) and combines results. So matching on any of these yields a match.</td>
</tr>
<tr>
<td>Location</td>
<td>location-dup</td>
<td>Uses Address Line 1 as the key or acquisition attribute and scores on other fields, using weights. Address Line 1 has the highest weight (about 67%).</td>
</tr>
</tbody>
</table>
Match Options Behavior

When finding out duplicate organization and person parties, some users might want to match potential duplicate records using multiple addresses and phones. However, indexing multiple records with multiple addresses and phones can adversely affect performance. The data quality functionality tries to strike a balance between the accuracy and speed of duplicate identification by taking the following optimal approach:

1. Partitioning the data based on customer data entities and their attributes in staging tables before loading the data into the IIR repository.
2. Providing appropriate match options on the configuration page.

The following table describes real time matching behavior for the different available match options.

<table>
<thead>
<tr>
<th>Entity</th>
<th>User Options</th>
<th>API or Web service Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Match Against:</td>
<td>For Primary Only match option, service goes against PRIMARY search def</td>
</tr>
<tr>
<td></td>
<td>Primary Only</td>
<td>For Primary OR All Addresses match option, service goes against ADDRESS search def, if address of the record being matched is not null, else goes against the PRIMARY search def</td>
</tr>
<tr>
<td></td>
<td>Primary OR All Addresses</td>
<td></td>
</tr>
<tr>
<td>Entity</td>
<td>User Options</td>
<td>API or Web service Behavior</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Person</td>
<td>Match Against:</td>
<td>For Primary Only match option, service goes against PRIMARY search def.</td>
</tr>
<tr>
<td></td>
<td>Primary Only</td>
<td>For Primary OR All Addresses match option, service goes against ADDRESS search def, if address of input record is not null, else goes against the PRIMARY search def.</td>
</tr>
<tr>
<td></td>
<td>Primary OR All Addresses</td>
<td>For Primary OR All Addresses OR All Phones match option:</td>
</tr>
<tr>
<td></td>
<td>Primary OR All Phones</td>
<td>If both Address and Phone of the record being matched are not null, the service goes against ADDRESS OR PHONE search def based on tiebreaker logic.</td>
</tr>
<tr>
<td></td>
<td>Primary OR All Addresses OR All Phones (with a user specified tiebreaker, when the record being matched has both address and phone)</td>
<td>If Address is not null, the service goes against ADDRESS search def.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If Phone is not null, the service goes against ADDRESS search def</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If both Address and Phone are null, the service would go against the PRIMARY.</td>
</tr>
</tbody>
</table>

The following table describes batch matching behavior for the different available match options.

<table>
<thead>
<tr>
<th>Entity</th>
<th>User Options</th>
<th>API or Web service Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Match Against:</td>
<td>For Primary Only match option, service goes against only ORG PRIM</td>
</tr>
<tr>
<td></td>
<td>Primary Only</td>
<td>For Primary AND All Addresses match option, service goes against both ORG PRIM and ORG ADDRESS.</td>
</tr>
<tr>
<td></td>
<td>Primary AND All Addresses</td>
<td></td>
</tr>
<tr>
<td>Entity</td>
<td>User Options</td>
<td>API or Web service Behavior</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Person</td>
<td>Match Against:</td>
<td>For Primary Only match option, service goes only against PER PRIM</td>
</tr>
<tr>
<td></td>
<td>Primary Only</td>
<td>For Primary AND All Addresses match option, service goes against both PER PRIM and PER ADDRESS.</td>
</tr>
<tr>
<td></td>
<td>Primary AND All Addresses</td>
<td>For Primary AND All Phones match option, service goes against both PER PRIM and PER PHONE.</td>
</tr>
<tr>
<td></td>
<td>Primary AND All Phones</td>
<td>For Primary AND All Addresses AND All Phones match option, service goes against all: PER PRIM and PER ADDRESS and PER PHONE.</td>
</tr>
</tbody>
</table>

**Real-Time Duplicate Prevention Components: How They Work Together**

Real-time duplicate prevention enables you to prevent the individual entry of duplicate entities, such as organization, person, or location, in the Oracle Sales Cloud registry, through data quality matching services consuming applications.

Real-time duplicate prevention comprises the following components:

- Real-Time Matching Service
- Data quality service consuming applications
- Matching configurations
- Server configurations

This figure shows the real-time duplicate prevention components in action.
Real-Time Matching Service
During the data entry process, the Real-Time Matching Service finds all possible duplicate records that may exist for an entered record, and assigns a match score to each potential duplicate identified.

Consuming Applications
When you try to enter a new person, organization, or location record, the data quality functionality finds all possible duplicate records and assigns a match score to each of them. Based on the match score and the threshold settings in the configuration, you are provided the option to either select an existing duplicate record or continue to create a new record.

Matching Configurations
Matching configurations are predefined configurations used for real-time and batch matching of party entities to prevent and identify duplicate entries. The data quality functionality has three predefined real-time matching configurations:

- Real Time Location Duplicate Prevention
- Real Time Organization Duplicate Prevention
- Real Time Person Duplicate Prevention

You can search, review, and edit these seeded matching configurations. While you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements.

Server Configurations
Real-time duplicate prevention makes use of the Real-Time and Batch Basic Match Server.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing, Server
configurations show all the parameters that can be set at either the matching configuration level or the server configuration level and provide the ability to modify the values of the server configuration level parameters. The parameters that are set at the server level are applicable to all the matching configurations.

**Batch Duplicate Identification Components: How They Work Together**

Batch duplicate identification enables you to identify duplicate entries in the registry, between registry and sets of data, such as import batches, or within sets of data to resolve them by merging or linking.

Batch duplicate identification comprises the following components:
- Batch Matching Service
- Data quality service consuming applications
- Matching configurations
- Server configurations

This figure shows the batch duplicate identification components in action.

**Batch Matching Service**

The Batch Matching Service is used for identification of potential duplicates for entities such as persons, organizations, and locations.

**Consuming Applications**

The batch duplicate identification capability finds use in master data management applications, and other data quality service consuming applications.
Matching Configurations
Matching configurations are predefined configurations used for batch and real-time matching of party entities with a view to identifying and preventing duplicate entries in the registry. The data quality functionality has three predefined batch matching configurations:

- Batch Location Basic Duplicate Identification
- Batch Organization Basic Duplicate Identification
- Batch Person Basic Duplicate Identification

You can search, review, and edit these seeded matching configurations. While you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suit your requirements.

Server Configurations
Batch duplicate prevention makes use of the Real-Time and Batch Basic Match Server.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing. Server configurations show all the parameters that can be set at either the matching configuration level or the server configuration level and provide the ability to modify the values of the server configuration level parameters. The parameters that are set at the server level are applicable to all the matching configurations.

Matching Configuration Parameters
Matching configuration parameters are system-level parameters that control aspects of the data quality matching services. The data quality matching services read values from the ZCQ_CONFIG_PARAM_VALUES and ZCQ_SO_PARAM_VALUES tables.

Matching configurations are predefined configurations that include parameters from both Oracle Sales Cloud applications and the integrated third-party data quality engine. Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suit your requirements. The text describes only the generic data quality parameters and their values. For third-party data quality engine specific parameters and their values, see relevant third-party documentation.

Real Time Location, Organization, and Person Duplicate Prevention Parameters
The following parameters control matching operations for preventing individual entry of duplicate entities, such as location, organization, and person through calling touch point applications into the registry.

MaxNumReturn
Parameter Value: 1 or more. Default Value: 10
Parameter Description: Controls the maximum number of matched records returned by the data quality real-time matching service.
ScoreThreshold
Parameter Value: Between 0 and 100. Default Value: 90
Parameter Description: Determines the score above which the matched records are returned by the matching service. Records equal to or greater than the score are considered as matches and the records with scores less than the threshold are rejected.

Batch Location, Organization, and Person Basic Duplicate Identification Parameters
The following parameters control matching operations for identification of duplicate entries in the registry, between registry and sets of data, such as import batches, or within sets of data to resolve them by merging or linking.

MatchMode
Parameter Value: Optimized or Exhaustive Default Value: Optimized
Parameter Description: Determines how the records in the batch are matched during batch matching process. Select Exhaustive to include in the batch match process even the records already showing up in previously matched groups. Select Optimized to skip such records.

NumChildJobs
Parameter Value: 1 or more. Default Value: 2
Parameter Description: Determines the number of child requests spawned for a batch matching request. The child requests are useful when there is a need to process data in parallel.

ScoreThreshold
Parameter Description: As described earlier.

Manage Cleansing Configurations

Cleansing Configurations: Explained

Cleansing configurations are predefined configurations used for real-time and batch cleansing of the location entity.
There are two predefined cleansing configurations, one each for real-time and batch matching:
• Real Time Data Quality Location Cleanse
• Batch Data Quality Location Cleanse
You can search, review, and edit these seeded matching configurations. Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements. These configurations include parameters from both Oracle Sales Cloud applications and the integrated address cleansing engine. While some of the parameters are single valued others are multi-valued.

Real-Time Cleansing

Real-time cleansing includes an online, interactive service to cleanse, standardize, and validate addresses during the data entry process either through a UI or any other service creating address data into the registry.
Batch Cleansing

Batch cleansing enables you to perform address cleansing, standardization, and validation for a subset or entirety of the location records in the registry, or as part of a data import process. It also enables you to ensure data accuracy of the registry. The need to cleanse the registry data arises owing to data decay over time and the need to resolve inherited issues from consolidated data in the registry. For example, postal codes and city boundaries change over time and require regular address cleansing to ensure that the addresses in the registry are correct, validated, and standardized at all times. Besides, historic incoming address data may have originated from multiple sources with each source system following different data storage formats and norms and the addresses might not have been cleansed during the import or consolidation process.

Real-Time Address Cleansing Components: How They Work Together

Real-time cleansing provides consuming applications an online, interactive service to cleanse, standardize, and validate addresses during the data entry process.

Real-time address cleansing comprises the following components:

- Real-Time Cleansing Service
- Data quality service consuming applications
- Cleansing configurations
- Server configurations

This figure shows the real-time address cleansing components in action.
**Real-Time Cleansing Service**

The Real-Time Cleansing Service performs validation, standardization, and cleansing of addresses during data entry into the registry.

**Consuming Applications**

Real-time address cleansing finds use in data quality service consuming applications. When a user tries to enter a new address record through the UI of these applications or through any other service creating address data into the registry, the service validates the entered address data against country-specific postal service lists, such as United States Postal Service. If there are any issues with the entered address data, the service returns a list of possibilities from which the consuming application user can select an appropriate one or continue to create a new address record.

**Cleansing Configurations**

Cleansing configurations are predefined configurations used for real-time and batch cleansing of the location entity.

The customer data quality application has a predefined real-time cleansing configuration named Real Time Data Quality Location Cleanse. The real-time address cleansing service uses this cleansing configuration to derive the default country to be used to perform validation when address parsing fails to find a valid country within the address.

You can review and edit this predefined cleansing configuration. While you cannot add or delete a configuration parameter, you can change the configuration name and update the parameter values to suit your requirements.

**Server Configurations**

Real-time cleansing makes use of the Real-Time Cleanse Server configuration.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

**Batch Address Cleansing Components: How They Work Together**

Batch cleansing enables you to perform address cleansing, standardization, and validation for a subset or entirety of the location records in the registry, or as part of a data import process.

Batch address cleansing comprises the following components:

- Batch Cleansing Service
- Data quality service consuming applications
- Cleansing configurations
- Server configurations
This figure shows the batch address cleansing components in action.

**Batch Cleansing Service**

The Batch Cleansing Service performs validation, standardization, and cleansing of addresses to ensure data accuracy during the import of addresses in batches, to reduce decay and obsolescence of existing address data in the registry, and to resolve inherited issues from consolidated data in the registry.

**Consuming Applications**

Batch address cleansing finds use in master data management applications, and other data quality service consuming applications that import data in batches. Unlike real-time cleansing that validates addresses one by one and runs in suggest cleansing mode, batch cleansing allows for validation and correction of multiple addresses through a single call.

**Cleansing Configurations**

Cleansing configurations are predefined configurations used for batch and real-time cleansing of the location entity.

The customer data quality feature has a predefined batch cleansing configuration named Batch Data Quality Location Cleanse. The batch address cleansing service uses this cleansing configuration to derive the default country to be used to perform validation when address parsing fails to find a valid country within the address.

You can review and edit this predefined cleansing configuration. While you cannot add or delete a configuration parameter, you can change the configuration name and update the parameter values to suite your requirements.
Server Configurations

Batch address cleansing makes use of the Batch Data Quality Location Cleanse server configuration.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

Cleansing Configuration Parameters

Cleansing configuration parameters are system-level parameters that control aspects of the data quality cleansing services. These services read values from the ZCQ_CONFIG_PARAM_VALUES and ZCQ_SO_PARAM_VALUES tables.

Cleansing configurations are predefined configurations that include parameters from both Oracle Sales Cloud and the integrated third-party data quality engine. Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements. The text describes only the generic data quality parameters and their values. For third-party data quality engine specific parameters and their values, see relevant third-party documentation.

Real Time Data Quality Location Cleanse Parameters

The following parameters control real-time data cleansing operations that happen at touch point applications.

RuntimeMapping
TRUE or FALSE TRUE
Determines if the real-time cleansing service should use the attribute mapping defined between the data quality application attributes and the third-party data quality engine. If the value is set to FALSE, the data quality real-time service will pass through attributes as they are between the data quality application and the third party cleansing engine.

Batch Data Quality Location Cleanse Parameters

The following parameters control address cleansing, standardization, and validation operations for a subset or entirety of the location records in the registry, or as part of a data import process.

AMCommitRate
1 or more 100
Controls the transaction batch size for the address cleansing process which in turn determines how often the cleansing results are committed. Set this value based on available system resources for optimum performance. If the value is too low, the data is committed often, which may cause slow performance. In contrast setting it to too high a value may require higher memory for processing.

NumChildJobs
1 or more 2
Determines the number of child requests spawned for a batch matching request. The child requests are useful when there is a need to process data in parallel.
Manage Data Synchronization

Data Synchronization: Explained

Real-time duplicate prevention during data entry and batch duplicate identification of entities in the registry and during the data import process require both initial indexing and index synchronization of the integrated third party data quality engine repository with the registry.

Periodic index synchronization of the third party data quality engine repository with the registry lets you account for the continual updates made to the registry.

Some of the entities that need to be synchronized to the matching engine repository are organization, person, and location. The synchronization should be done when these entities are created or updated.

However, no data synchronization is required for advanced batch match configuration and real-time and batch cleanse operations because of the following reasons:

- Advanced configuration is used only for batch matching such as finding duplicates within a set of interface tables records before running a bulk import process to load the records in the registry.
- Data synchronization is not needed for real-time address cleansing functions as the address is checked against the country specific postal address files and not against the existing address in the registry. Note that the postal reference files need to be updated periodically.
- During the batch cleanse operation, records in a batch are sent to the data quality server for cleansing one at a time and the data quality engine returns the cleansed record to the registry likewise, one at a time, thus completing the loop.

Synchronizing Registry and Data Quality Engine Repository Data: Worked Example

This example demonstrates the index synchronization of the integrated third party data quality engine repository with the registry. Index synchronization facilitates accounting for the changes made to the registry as part of creating new and updating existing organization, person, and location records since initial indexing. This example focuses on the index synchronization of person party records for the third-party data quality engine Informatica Identity Resolution (IIR).

Index synchronization for the Informatica Identity Resolution data quality engine involves the following tasks:

- Scheduling the synchronization process: Schedule this task to run periodically to account for the continual updates to the registry. While running the synchronization process you can select the entities for which
you want to synchronize updates and specify the date and time from which to synchronize the updates. A synchronization request performs index synchronization from the last synchronized date and time until the date and time it is submitted.

- Starting Informatica Identity Resolution (IIR) Update Synchronizer: Administrators and integrators typically perform this task. Once started the synchronizer can be left running, for the life cycle of the Informatica Identity Resolution server, unless there is a need to bring it down as part of a maintenance window. However, whenever there is a loss of connectivity with the database Update Synchronizer stops functioning and needs to be manually restarted.

### Note

Initial indexing and index synchronization are required only for performing matching operations aimed real-time duplicate prevention and batch duplicate identification. Real-time and batch cleanse operations do not require initial indexing and Index synchronization.

### Running the Schedule Synchronization Process

1. Navigate to Setup and Maintenance from the Tools menu.
2. Search for the Manage Data Synchronization task.
3. Click the Go to Task icon.
4. Select Refresh Identity Table Information option from the Actions menu on the Manage Data Synchronization page.
5. Enable all the relevant identity tables (for this example PER_PRIMARY_IDT, PER_ADDRESS_IDT, PER_PHONE_IDT) by selecting Enable for Sync, check box.
6. Enter the following Synchronization Options for each identity table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>Leave blank when using the predefined system, FusionDQRealtime, for all countries. If using country specific system, enter the country, such as US.</td>
</tr>
<tr>
<td>Last Synchronized Time</td>
<td>Select Date and Time: 3/23/11 9:30:15 AM</td>
</tr>
</tbody>
</table>

7. Click Save.
8. Click the Schedule Synchronization Process button.
9. Click Advanced.
10. Select Using a Schedule. From the Frequency drop-down list, select Hourly/Minute and provide a value for the frequency. For example, 5 minutes.
Note
The frequency needs to be determined by the business requirement.

11. From the Calendar, select the End Date.

Note
The end date needs to be determined by the business requirement.

12. Submit the scheduled Sync ESS Job.
13. Click OK to return to the Manage Data Synchronization page.
14. Hover on the Process Status field corresponding to each relevant identity table to know the status of the child synchronization request spawned for that table.
15. Click Save and Close.

Starting Informatica Identity Resolution Update Synchronizer
1. Sign in the Informatica Identity Resolution (IIR) host machine.
2. Enter cd <PROV.TOP>/InformaticaIR/bin
3. Enter setfusionEnv.sh
4. Start the IIR console client using the admin option, ./idsconc -a
5. Select Run Synchronizer on the Tools menu to launch the synchronizer.
6. In the Update Synchronizer dialog, select All as the value for IDT Name, use the default values for the rest of the fields, and click OK.
7. Verify that the updated and newly created person records are available in IIR, by searching for persons in the Per-dup tab of IIR Web Search Client.
8. Sign out of the Informatica Identity Resolution (IIR) user interface.

Synchronizing and Optimizing Database Search Indexes for CRM Objects: Explained

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

Synchronizing Database Search Indexes

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

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

To schedule the database synchronization job:

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

2. Click the Schedule New Process button.

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

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

Optimizing Database Search Indexes

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

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

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

To schedule the database optimization job:

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

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

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

5. Click OK.
FAQs for Define Data Quality

What's the difference between different matching configurations and matching server configurations?

Matching configurations are used for real-time and batch matching of party entities with a view to preventing and identifying duplicate entries in the registry. The matching configuration includes the parameters that can be set at the matching configuration level and can be modified depending on matching strategy, data, and result requirements. Use the matching configuration to view or change the values of the matching configuration level parameters.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing. Server configurations show all the parameters that can be set at either the matching configuration level or the server configuration level along with the parameter type and the cardinality of the parameter. The parameters that are set at the server level are applicable to all the matching configurations. Use the server configuration to view or change the values of the server configuration level parameters.

What's the difference between different cleansing configurations and cleansing server configurations?

Cleansing configurations are used for real-time and batch cleansing of addresses during data entry and on a periodical basis to cleanse, standardize, and validate existing addresses to ensure data accuracy. The cleansing configuration includes the parameters that can be set at the cleansing configuration level and can be modified depending on cleansing strategy, data, and result requirements.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

What's the difference between real-time duplicate prevention and duplicate identification?

Real-time duplicate prevention involves identification of all possible duplicate records that may exist in the registry for an entered record. This enables
prevention of individual entry of duplicate entities, such as organization, person, or location, through calling touch point applications in the registry.

Duplicate identification of entities is achieved by batch matching. The objective is to identify potential duplicate entities already existing in the registry, and then resolve the duplicates by merging or linking.

**What's the difference between basic duplicate identification and advanced duplicate identification?**

The objective of basic duplicate identification is to resolve the duplicate entities in the registry by merging or linking.

Advanced duplicate identification is mainly done to find duplicates in a group of records before uploading them to the registry. For example, an acquiring company may do an advanced duplicate identification of the legacy data of the acquired company before uploading the data to the registry.

**What's the difference between real-time address cleansing and batch address cleansing?**

Real-time address cleansing is an online, interactive service to cleanse, standardize, and validate addresses during the data entry process.

Batch address cleansing enables you to cleanse, standardize, and validate addresses that are either existing in the registry or are being imported into it.
Define Customer Hub Configuration

Run Request Dispatch Job

Configuring the Data Quality Management Request Dispatcher: Explained

Use the request dispatcher to process cleansing and duplicate identification batches, and duplicate resolution requests. The Run Request Dispatcher process can be scheduled to run as one time process or at a repeat schedule. This task can also be used to monitor the cleansing and duplicate identification batches and duplicate resolution requests.

You can configure the request dispatcher to run in the following two modes:

- Basic
- Advanced

Running the Dispatcher in the Basic Mode

Run the dispatcher in this mode to submit the request dispatcher job for immediate processing of the duplicate identification batches, cleansing batches, or resolution requests.

Running the Dispatcher In the Advanced Mode

This mode lets you schedule the request dispatcher job to be run either immediately or at a specified intervals, such as, every 15 minutes to process duplicate identification batches, cleansing batches, or resolution requests.

Manage Customer Hub Profile Options

Customer Hub Profile Options: Explained

The Customer Hub profile options provide data access and processing options related to data governance, duplicate identification processes, data cleansing processes, and duplicate resolution requests.
**Note**
The Hub profile option values can be set only at the site level using the predefined profile option definitions.

Navigate to Manage Customer Hub Profile Options page using the Setup and Maintenance option on the Tools menu to configure the following profile options:

- Address Cleansing Configuration
- Data Cleansing Process Batch Size
- Resolution Request Type Default
- Duplicate Identification Organization Configuration
- Duplicate Identification Person Configuration
- Duplicate Identification Process Batch Size
- User Merge Requests

**Address Cleansing Configuration**
Specify the data quality address cleansing configuration used by the data cleansing process.
Default value: BT_LOC_CLEANSE

**Data Cleansing Process Batch Size**
Control the transaction batch size for the data cleansing process. Set this value based on available system resources. Set this value based on available system resources.
Default value: 100

**Resolution Request Type Default**
Specify the default request type value for duplicate resolution requests.
Default value: Merge

**Duplicate Identification Organization Configuration**
Specify the data quality organization match configuration used by the duplicate identification process.
Default value: BT_ORG_DUP_BASIC

**Duplicate Identification Person Configuration**
Specify the data quality person match configuration used by the duplicate identification process.
Default value: BT_PERSON_DUP_BASIC

**Duplicate Identification Process Batch Size**
Control the transaction batch size for the duplicate identification process. The value is used to chunk records in the batch and process each chunk in a loop as a separate transaction.
Default value: 100

**User Merge Requests**

Specify the processing of merge requests, such as, process without approval and approval needed. The different values of this profile option are explained below.

- **Allow Processing Without Approval**: If the User Merge Requests profile option is set to this value, user merge requests are processed immediately without approval by Data Steward.
- **Process Subject to Approval**: If the User Merge Requests profile option is set to this value, user merge requests are entered for review by Data Steward, who can decide to process them or not.
- **Unspecified (NULL)**: This default value of the User Merge Requests profile option indicates that the Customer Hub is not configured and that user merge requests can be processed immediately without the need for Data Steward approval.

**Note**

The default value of the User Merge Requests profile option, Unspecified or NULL, will enable the user to submit the merge request to be processed immediately without the need for Data Steward approval only if the user has the Submit Trading Community Merge Request privilege in addition to the Enter Trading Community Merge Request privilege.

Sales accounts merges initiated from the Customer Center, or the Enter Merge Request setup task, or the Automerge web service, DuplicateResolutionRequestService, cannot happen immediately without Data Steward processing. Note that to allow Data Steward processing the User Merge Requests Profile Option should be set to Process Subject to Approval, otherwise the merge requests will error.

**Define Survivorship**

**Manage Survivorship Rules**

**Survivorship Rules: Explained**

Survivorship rules are a collection of business rules defined for selecting master record and attributes to be retained during merge or link, or update operations.

Survivorship rules facilitate intelligent creation of the best version record from multiple source systems, based on business rules. You can configure survivorship rules to determine how conflicts should be resolved while merging or updating duplicate records from different source systems to create the best version record. You can specify criteria to determine the master (or the surviving) party record for duplicate resolution. The survivorship rules criteria determine the attributes of the best version record that should be retained from a particular source system. The rules use the priority assigned to different source systems, as well as other criteria to determine which values to retain for certain attributes.
Managing Survivorship Rules

Use the UI available from the Manage Survivorship Rules Setup and Maintenance task to view, create, edit, and delete survivorship rules. The survivorship functionality is disabled by default. To enable survivorship, set the ZCH_ENABLE_SURVIVORSHIP profile option to Yes using the Manage Customer Hub Profile Options Setup and Maintenance task.

Importing and Exporting Survivorship Rules

Use the Import Survivorship Rules Dictionary web service to import the survivorship rules dictionary XML file created or stored elsewhere into the application. Use the Export Survivorship Rules Dictionary web service to export survivorship rules dictionary from the application.

Defining Survivorship Rules: Worked Example

This example demonstrates how to create a survivorship rule. Survivorship rules enable intelligent creation of the best version record, especially from multiple source systems, by specifying criteria for selecting the record to be retained during a merge operation.

Use the UI available from the Manage Survivorship Rules Setup and Maintenance task to create a survivorship rule and to specify the criteria for selecting the master record.

Create Survivorship Rule

1. Navigate to the Manage Survivorship Rules page.
2. Click Create.
3. Enter the following sample information.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule Name</td>
<td>PickPersonMasterRule</td>
</tr>
<tr>
<td>Description</td>
<td>Select the master person record based on original source system of the record.</td>
</tr>
<tr>
<td>Rule Type</td>
<td>Set_Master</td>
</tr>
<tr>
<td>Object Type</td>
<td>Person</td>
</tr>
</tbody>
</table>

Note

You can create a survivorship rule for the following two types of party records: person and organization.
4. Click Apply. You are taken to the Define Survivorship Rules: Select Master Record page.

**Specify Criteria for Selecting the Master Record**

1. On the Define Survivorship Rules: Select Master Record page, enter the following IF/THEN rules condition.

<table>
<thead>
<tr>
<th>Rule Condition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF Condition</td>
<td></td>
</tr>
<tr>
<td>THEN Condition</td>
<td></td>
</tr>
</tbody>
</table>

**Note**

If survivorship rules conflict with a real life situation, an error message is displayed and the merge fails. In the context of this example, the survivorship rule in use states that if a person party record is from the source system GSI, it should be set as the master record. Consequently, if at runtime a merge request has more than one person record from GSI, the survivorship engine returns multiple master records, which is incorrect. In such a situation, an error message is displayed stating that the merge request failed because conflicting survivorship rules are defined and that the administrator should modify the rules to ensure that only one master record is returned.

2. Verify that the effective date of the rule is set to the rule’s creation date and, if required, modify it as appropriate.

3. Select Medium as the priority for the rule.

4. Select the Rules Active option.

5. Click Save or Save and Close.

6. Click Submit.

**FAQs for Manage Survivorship Rules**

**What's the difference between survivorship rules and merge agreement rules?**

Survivorship rules are a collection of business rules defined for the creation of the best version record intelligently, especially from multiple source systems, by specifying criteria for selecting record and attributes to be retained during merge or link, or update operations.

A merge agreement rule is a collection of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Merge requests that violate these rules are either rejected or end in error.
How can I check if the survivorship rules that I created work?

To verify whether survivorship rules are active, you need to create two parties and merge them into a single record. If your survivorship rules are active, you can see them work, as the attributes that satisfy the survivorship rules are selected.

**Note**

Before you check your survivorship rules, ensure that the survivorship rules you created are active.

To check your survivorship rules:

1. Go to the Parties page and create two organizations that you can merge into one: Pinnacle Flowers Old Org and Pinnacle Flowers New Org, for example.
2. Create a merge request from the **Setup and Maintenance, Enter Merge Request** page.
3. Submit the merge using the **Setup and Maintenance, Run Request Dispatch Job** page.
4. Check the status of the merged records using the **Navigator, CDM, Duplication Resolution** page.
5. Check the attributes in the merged organization to ensure that your survivorship rules work.

What are the seeded sample survivorship rules that Oracle Fusion Applications provide by default?

Oracle Fusion Applications offer the following sample survivorship rules:

- For organizations, history wins: The attribute value with earlier source update date wins. This rule is applicable for all organization attributes.
- For persons, recent wins: The attribute value with later source update date wins. This rule is applicable for all person attributes
- For organizations, assign the highest source confidence level to the DUNS number.

The seeded survivorship rules are in inactive status out-of-the-box. These rules are updated with every release. However, once you change these rules, they are not updated.

**Manage Source System Confidence**

**Source System Confidence Levels: Explained**

Oracle Fusion Master Data Management Applications enable you to combat problems related to incomplete data across source systems by offering the ability to create a single best-of-breed master record. When there are multiple sources containing attributes related to the same customer record, getting reliable data
becomes one of the greatest challenges to creating a best-of-breed record. Source system confidence levels enable you to indicate the reliability of a particular source system for specific attributes. You can set a source confidence score per source system for each attribute in both Person and Organization object types. These scores become deciding factors in selecting survivor attribute values during merge, insert, or update operations.

You can change the confidence levels associated with source systems at any point, and the best-of-breed record gets updated based on the new confidence levels. You can also track source system confidence level changes between specific dates.

FAQs for Manage Source System Confidence

**How do source system confidence levels work?**

Assign source confidence levels for specific attributes in source systems based on your understanding of the quality of the data stored in the various source systems within your organization. Once you have assigned source system confidence levels for the attributes you need, these attributes will be selected from the concerned source systems based on the survivorship rules that you set up for them.

Source system confidence levels range from 0 to 100, and are set by default to 0. This means that data attributes from any source system can be included in the best-of-breed records as long as there are no other competing attributes in other source systems. When all source system confidence levels are set to 0, all conflicts are listed out and made available to the Master Data Applications Administrator, who then must decide upon the source system whose attributes must be included into the best-of-breed records.

**Can I apply the same source system confidence level to multiple attributes of an object type simultaneously?**

Yes. Manage source system confidence levels using the Setup and Maintenance, Manage Source System Confidence page. To apply the same source confidence level to multiple attributes of an object type, select the object type and, from the list of available attributes, select the attributes to which you want to apply the new confidence level. Add the source system that you want to associate with the attributes you selected. You can now specify one confidence level for each of these attributes in the source system.

You can similarly associate other source systems with the attributes you selected and specify new confidence levels as required.

**What are Set Master and Set Attribute rules?**

Set Master rules enable you to define the logical business criteria for selecting the master record from a set of potential duplicate records for merge operations.

Set Attribute rules enable you to define the logical business criteria for selecting the best attribute values from multiple input records during merge and insert or update operations.
Manage Party Center Trees

Party Center: Explained

The party center functionality enables a comprehensive management of party information. A party is a physical entity, such as a person, organization or group, that the deploying company has an interest in tracking. The party center functionality lets you bring a particular party into context, fetch data from various systems about the party in context, and review and edit the data in one location.

The party center enables you to do the following:

- Create Organizations
- Manage Organizations
- Create Groups
- Manage Groups
- Create Persons
- Manage Persons

Party Center Tree

You can select a party from the organizations, person, or groups lists to bring it into context. Once a party is in context, you can fetch, review, and edit its data in the party center. The party entities are displayed as nodes in the party center tree. A party center tree comprises nodes that let you fetch data related to the party in context from various systems, and review and edit the data displayed in each node.

The party center nodes display information, such as profile, profile history, party usage assignments, relationships, classifications, source system references, and customers associated with the party in context.

Party Center Trees: Explained

A party center tree comprises nodes that let you fetch data about the party in context from various systems, and review and edit the data displayed in each node. You can also customize a party center tree by showing or hiding its various nodes as required, reviewing and editing the node names as appropriate, and determining the need for entering additional parameter values to gain access to a node.

There are three types of party center trees:

- Organization: An organization party tree displays nodes representing organization parties across usages.
• Group: A group party tree displays nodes representing group parties across usages.

• Person: A person party tree displays nodes representing person parties across usages. All nodes on the person party tree are visible to all users.

Each tree displays slightly different nodes, and the information that you can view and edit on each node depends upon your security privileges.

A party center tree may include the following nodes:

• Profile
• Profile History
• Usage assignments
• Relationships
• Classifications
• Source system references
• Linked parties

Note
The profile history information for a group party type is displayed in the Profile node.

Some of them are explained below:

Profile

This node lets you review and edit party profile information such as name, party usage, additional names, additional identifiers, addresses, and contact points.

Profile History

This node lets you view the history of changes to profile records of the party, person or organization, in context in the party center.

Usage Assignments

This node launches a task flow that gives visibility into what party usages have been currently assigned to the party. Party usages that allow manual create, manual update, and unconditional assignment are maintainable through this user interface.

Relationships

This node lets you view, create, edit, and delete all relationships for the party in context in the Party Center. The node also displays the contact details of a relationship.

Classifications

This node lets you view, create, edit, and delete classifications for the party in context in the Party Center. The displays the primary classification first in the list of classifications.
**Source System References**

This node lets you view, create, edit, and delete all source system references contributing to the master organization party record in context in the Party Center.

**Linked Parties**

This node lets you manage the parties linked to the party in context.

**Managing Party Center Trees: Explained**

Use the Setup and Maintenance option on the Tools menu to navigate to the Define Customer Hub Configuration tasklist. Go to the Manage Party Center Tree task to manage a party center tree. Alternatively, use the Actions menu options in the regional area of Oracle Sales Cloud to personalize a party center tree.

Specify the following attributes for each node in the party center tree:

- **Name**
- **Visible**
- **Parameter**

**Name**

Set this attribute to specify the name of the node that you want to display in the party center tree.

**Visible**

Set this attribute to specify the display status, hidden or visible, of a node in the party center tree.

**Parameter**

Specify the additional parameter values required to gain access to a node.

**Manage Customer Data Hub Notes**

**Defining Notes: Points to Consider**

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:
• Note Types
• Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn't exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article Extending Oracle Sales Cloud Applications: how it works.

Manage Agreement Rules

Agreement Rules: Explained

An agreement rule is a collection of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Merge requests that violate these rules are either rejected or end in error.

Agreement rules let you check a merge request for any veto conditions that can prevent a merge from occurring. These rules save resources and time by obviating the need to review merge request to prevent undesired merge from being processed. Besides, agreement rules prompt you to consider alternative duplicate resolution mechanism such as linking.

Agreement rule can be of the following two types:

• Predefined
• User-defined
### Predefined Agreement Rules

These are agreement rules that are predefined in the application. You can only view the predefined agreement rules.

The following table describes the predefined agreement rules for the Oracle Sales Cloud. Merge requests that violate these rules are automatically rejected.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HR_APPLICANT_VETO</td>
<td>Prevents a person party with an active party usage of HR_APPLICANT from merging with another party.</td>
</tr>
<tr>
<td>HR_EMPLOYEE_VETO</td>
<td>Prevents a person party with an active party usage of HR_EMPLOYEE from merging with another party.</td>
</tr>
<tr>
<td>HR_CONTINGENT_WORKER_VETO</td>
<td>Prevents a person party with an active party usage of HR_CONTINGENT_WORKER from merging with another party.</td>
</tr>
<tr>
<td>HR_PARTY_SITE_VETO</td>
<td>Prevents a party site with active original system reference from Fusion HCM system from merging with another party site.</td>
</tr>
<tr>
<td>RESOURCE_PERSON_VETO</td>
<td>Prevents a person party with an active party usage of RESOURCE from merging with another party.</td>
</tr>
<tr>
<td>BANK_VETO</td>
<td>Prevents an organization party with an active party usage of BANK from merging with another party.</td>
</tr>
<tr>
<td>CLEARINGHOUSE_VETO</td>
<td>Prevents an organization party with an active party usage of CLEARINGHOUSE from merging with another party.</td>
</tr>
<tr>
<td>BANK_BRANCH_VETO</td>
<td>Prevents an organization party with an active party usage of BANK_BRANCH from merging with another party.</td>
</tr>
<tr>
<td>BRANCH_CLEARINGHOUSE_VETO</td>
<td>Prevents an organization party with an active party usage of CLEARINGHOUSE_BRANCH from merging with another party.</td>
</tr>
<tr>
<td>LEGAL_EXTLEGAL_PARTYUSGRULE_VETO</td>
<td>Prevents an organization party with active party usage of LEGAL_ENTITY from merging with another organization party with active party usage of EXTERNAL_LEGAL_ENTITY.</td>
</tr>
<tr>
<td>IC_PARTICIPANT_PERSON_VETO</td>
<td>Prevents a person party with an active party usage of INCENTIVE_COMP_PARTICIPANT from merging with another party.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IC_PARTICIPANT_ORG_VETO</td>
<td>Prevents an organization party with an active party usage of INCENTIVE_COMP_PARTICIPANT from merging with another party.</td>
</tr>
<tr>
<td>PARTNER_VETO</td>
<td>Prevents an organization party with an active party usage of PARTNER from merging with another party.</td>
</tr>
<tr>
<td>INACTIVE_PARTNER_VETO</td>
<td>Prevents an organization party with an active party usage of INACTIVE_PARTNER from merging with another party.</td>
</tr>
<tr>
<td>CUST_CONTACT_DIFF_RESOURCE_ORG_VETO</td>
<td>Prevents two partner owned contacts belonging to different resource organizations from being merged.</td>
</tr>
<tr>
<td>CUST_CONTACT_INTERNAL_PARTNER_VETO</td>
<td>Prevents a partner owned contact that does not belong to a resource organization from merging with another partner owned contact that belongs to a resource organization.</td>
</tr>
<tr>
<td>CUST_CONTACT_INTERNAL_PARTNER_ORG_VETO</td>
<td>Prevents a contact owned by an internal user from merging with a contact owned by a partner user.</td>
</tr>
<tr>
<td>CARD_ISSUER_VETO</td>
<td>Prevents an organization party with an active party usage of CREDIT_CARD_PROVIDER from merging with another party.</td>
</tr>
<tr>
<td>LEGAL_ENTITY_VETO</td>
<td>Prevents an organization party with an active party usage of LEGAL_ENTITY from merging with another party.</td>
</tr>
<tr>
<td>ESTABLISHMENT_VETO</td>
<td>Prevents an organization party with an active party usage of ESTABLISHMENT from merging with another party.</td>
</tr>
<tr>
<td>HIERARCHY_CYCLE_PREVENTATION_VETO</td>
<td>Prevents merge from happening when the master party record is at a level lower than the nonmaster party record within the same active tree version.</td>
</tr>
<tr>
<td>NAMEDACCOUNT_UNNAMEDACCOUNT_VETO</td>
<td>Prevents merge from happening when the master party record is not a named account whereas the nonmaster party record is a named account and these are in active hierarchies.</td>
</tr>
</tbody>
</table>

**User-defined Agreement Rules**

These are additional agreement rules that you can define to determine whether a merge request should be vetoed by the application. You can create, view, update, and delete user-defined agreement rules.
Defining Agreement Rules: Worked Example

This example demonstrates how to create user-defined agreement rules that you can use to prevent a merge request from being processed.

Agreement rules are collections of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Perform the following tasks to define agreement rules:

- Select the relevant agreement rules dictionary
- Add a new agreement rule


Select Agreement Rules Dictionary

An agreement rules dictionary is a collection of predefined terms and attributes for which agreement rules can be defined. Use the following steps to select the relevant agreement rules dictionary:

2. Select the relevant agreement rules dictionary from the Select Dictionary drop down list and click Load to load its predefined terms.
3. Select Visible for the terms and term attributes that you want to be available when defining rules later on in the flow.
4. Click Save or Save and Close.

Add a New Agreement Rule

2. Add a new rule and provide a rule name.
3. Click on Define Pattern.
4. In the Justification Reason, enter the reason for creating the agreement rule.
5. In the Justification Area enter the scope of the agreement rule.
6. Add a new pattern and complete the fields, using the sample information provided in this table. Use the default values except where indicated. Note that the relation is always AND between patterns and cannot be edited.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Dictionary Term</th>
<th>Term Alias</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>for each case where</td>
<td>PersonPartyVO</td>
<td>Person</td>
<td>AND</td>
</tr>
<tr>
<td>Pattern</td>
<td>Dictionary Term</td>
<td>Term Alias</td>
<td>Relation</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------</td>
<td>----------------</td>
<td>----------</td>
</tr>
<tr>
<td>for each case where</td>
<td>OrganizationPartyVO</td>
<td>NonmasterParty</td>
<td>AND</td>
</tr>
<tr>
<td>there is a case where</td>
<td>PartyUsageAssignmentVO</td>
<td>PartyUsageAssignment</td>
<td>AND</td>
</tr>
</tbody>
</table>

7. To define term conditions for each pattern defined, select the pattern in the Patterns table and navigate to the Conditions table underneath.

8. Add a new condition and complete the fields using the sample information provided in this table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Term Attribute</th>
<th>Operator</th>
<th>Value</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person. PartyNumber - missing PartyNumber</td>
<td>is not</td>
<td>1234</td>
<td>AND</td>
</tr>
<tr>
<td>NonmasterParty. MergeType</td>
<td>=</td>
<td>Nonmaster</td>
<td>AND</td>
</tr>
<tr>
<td>UsageAssignment. PartyUsageCode</td>
<td>=</td>
<td>HR_APPLICANT</td>
<td>AND</td>
</tr>
</tbody>
</table>

9. Click Save or Save and Close.

10. Click Submit.

**Enter Merge Request**

**Manual Merge: Explained**

Manual merge involves the selection of duplicate records directly by the user. This is unlike the system merge in which duplicates are selected using a matching configuration.

Manual merge is unlike the system merge in which duplicates are selected automatically using a matching configuration. Another difference is that while only two records can be merged as part of manual merge, multiple records can be merged as part of system merge.

Manually merge the two records that you think are duplicates as in one of the following two ways:

- Immediate manual merge
Deferred manual merge

**Immediate Manual Merge**

Involves specification of one of the two duplicate records as the master record for an immediate merge. In an immediate merge, while you can specify some of the attributes from individual records to be mapped to the master record, other attributes are automatically transferred.

**Deferred Manual Merge**

Involves submission of merge request to be reviewed and processed later by a data steward.

**Merging Records Manually: Worked Example**

This example demonstrates how you can manually merge two duplicate records. This example illustrates only the generic process for merging records using the Enter Merge Request task. You can also merge two duplicate records from certain other UI pages.

Merging records manually involves the following tasks:

- Identifying duplicate records
- Selecting the master and duplicate records
- Submitting the merge

**Identifying Duplicate Records**

1. Navigate to the Search and Select: Record to Merge page as follows: Set Up Tasks Enter Merge Request Search and Select: Record to Merge.
2. To search and select the record to be merged, complete the search criteria using the sample information provided in the given table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Name</th>
<th>Registry ID (Optional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>15918</td>
</tr>
</tbody>
</table>

3. Click Search.
4. From the search results, select the first record to be merged.
5. Select Merge Record from the Actions menu

**Selecting the Master and Duplicate Records**
The function privilege Enter Trading Community Merge Request controls whether the **Merge** secondary window appears at all.

1. In the **Merge** secondary window, specify the selected record either as master or duplicate.

2. To select the second (master or duplicate) record to be merged, in the **Party Picker** enter the sample information provided in the given table. Use the default values except where indicated.

<table>
<thead>
<tr>
<th>Name</th>
<th>Registry ID (Optional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>10854</td>
</tr>
</tbody>
</table>

3. Click Search.

4. From the search results, select the second record to be merged and click OK.

5. In the Merge secondary window, click Merge.

**Submitting the Merge**

1. On the Map Profile Attributes page, specify the attributes from each of the two records, such as name and gender, that should be mapped to the master record. Any child entities, such as addresses, accounts, account addresses, relationships, and contact points, are transferred to the master record.

2. Click **Submit**. The merge happens immediately if the data governance profile: User merge requests is set to Allow Processing Without Approval. If the profile is set to Process Subject to Approval, the merge is reviewed and processed later by a data steward.
Define Custom Enterprise Scheduler Jobs

Managing Job Definitions: Highlights

Oracle Enterprise Scheduler jobs are run in Oracle Fusion Applications to process data and, in some cases, to provide report output. A job definition contains the metadata that determines what the job does and what options are available to users who run the job. You can create and maintain job definitions for use in Oracle Fusion Applications.

Managing job definitions is described in the Oracle Fusion Applications Administrator's Guide and Oracle Fusion Applications Extensibility Guide for Developers. As you read content from these guides, note that the guides mention managing Oracle Enterprise Scheduler, including job definitions, from Oracle Enterprise Manager Fusion Applications Control. You can also access job definitions by starting in the Setup and Maintenance Overview page and searching for the Enterprise Scheduler job tasks for your applications.

Note
Oracle Enterprise Manager Fusion Applications Control is not available for Oracle Cloud implementations.

Selecting the Appropriate Implementation Task
Each Enterprise Scheduler job definition task uses one Java EE application, which is referenced in the task name. You must use the right task because, to access the product job definition that you want to view or work on, the view objects must be contained in the application. If you do not select the right task, then the job definition will not be displayed properly or function correctly. The application name is usually the same as the product that the job definition belongs to, but not always.

- For example, the Oracle Fusion Payables Java EE application contains the Oracle Fusion Expenses product view objects. To create or maintain a job definition for use in Expenses, you select the Manage Custom Enterprise Scheduler Jobs for Payables and Related Applications task.

- In another example, the Oracle Fusion Payments product view objects are contained in both Oracle Fusion Payables and Oracle Fusion Receivables
Java EE applications. You need to select the task appropriate to the job definition for Payments. Use the Manage Custom Enterprise Scheduler Jobs for Receivables and Related Applications task if the job is for receivables functionality, or the Manage Custom Enterprise Scheduler Jobs for Payables and Related Applications task if the job is for payables functionality.

- Use the task description to see the products that correspond to the Java EE application specified in the task name. For example, the description for the Payables task lists Oracle Fusion Payables, Assets, Expenses, and Payments.

- You can view task descriptions in the help window for the task, if any, or in the generated setup task lists and tasks report from the Getting Started page.

- If you have access to the Manage Task Lists and Tasks page, you can also open the details for specific tasks to see the description.

- For general information about product and Java EE application relationships, use Oracle Enterprise Manager Fusion Applications Control (Fusion Applications Control).

See: Topology Section

Viewing, Creating, and Editing Job Definitions

- You can access predefined and custom job definitions. In the table on the Manage Job Definitions tab, the Name column displays an asterisk for predefined job definitions. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Viewing Job Definitions

- You or a technical administrator can create jobs based on Java, PL/SQL, Oracle Business Intelligence (BI) Publisher, or any other supported technology. Every predefined or custom job must have a job definition. For Oracle Cloud implementations, custom job definitions can be created only for custom jobs based on Oracle BI Publisher reports. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Creating a Job Definition

- If you are using the Setup and Maintenance work area, then the Enable submission from Enterprise Manager check box is available for the job definition.

  - If you do not select this check box, then the job cannot be run from Enterprise Manager.
  
  - If you select this check box, then you can define parameters for this job definition only in Enterprise Manager. Save the rest of your work on the job definition, and then go to Enterprise Manager if you need to define parameters.

- You can edit all aspects of custom job definitions. For predefined job definitions, you can’t update parameters, but you can determine if user properties are read-only or not. You can also edit certain aspects of predefined definitions, which are described as job properties in the Oracle Fusion Applications Extensibility Guide for Developers.
Managing List of Values Sources: Highlights

A list of values source for Oracle Enterprise Scheduler job definitions determines where a list of values comes from and what the specific values are. These lists of values are used in parameters and application defined properties of job definitions. For example, you can use a source of country values for a Country job parameter.

Note
Since parameters for predefined job definitions cannot be edited, list of values sources are only for parameters in custom job definitions.

Managing list of values sources is fully described in the Oracle Fusion Applications Administrator’s Guide. As you read content from that guide, note that the guide describes managing Oracle Enterprise Scheduler, including list of values sources, from Oracle Enterprise Manager Fusion Applications Control. You can also access list of values sources by starting in the Setup and Maintenance Overview page and searching for Enterprise Scheduler job tasks.

Registering and Searching for List of Values Sources

- Create list of values sources to register them for use in job definitions.
  See: Registering Sources for Lists of Values

- Search for list of values sources to edit or delete, or to make sure a particular source does not already exist before you create it.
  See: Searching for List of Value Sources

Managing Job Sets: Highlights

Oracle Enterprise Scheduler job set definitions contain metadata that identify the jobs to be included in a single submission. Instead of submitting jobs separately to process data, and in some cases, generate output, users can submit one job set that contains the desired jobs. The job set definition also determines if the jobs run in serial or parallel, or based on other predetermined logic.

A job set can contain any number of individual jobs as well as other job sets. There can also be multiple levels of nested job sets within a single job set. For example, a job set can include three jobs and two job sets, one of which contains another job set.

Managing job sets is fully described in the Oracle Fusion Applications Administrator’s Guide.

Creating and Editing Job Sets

- The Oracle Fusion Applications Administrator’s Guide describes accessing job sets from Oracle Enterprise Manager Fusion Applications Control. You can also access job sets by starting in the Setup and Maintenance Overview page and using the Enterprise Scheduler job tasks for your applications.
FAQs for Define Extensions for Customer Data Management

What's the difference between Page Composer and Application Composer?

Page Composer is a web-based tool you can use to modify user interface (UI) pages and components for all products designated for use with Page Composer. Page Composer uses two different modes of Design View. The first mode, Design View: Standard mode, is selected by default in all Sales Cloud pages when opening a page with Page Composer with the Design button selected. The second mode, Design View: Direct Selection mode, is activated when you click the Select tab for the UI page you want to customize. In Sales Cloud, Direct Selection mode is available when you customize pages, but not when you personalize a dashboard page. With the Design View: Direct Selection mode, you can select and edit UI elements such as form fields and table columns. In Direct Selection mode, selectable UI components become apparent when you move your cursor over the UI component. Selectable UI components are highlighted and can be edited.

The following table describes how you can use each mode of Page Composer to customize dashboard pages and other select pages (such as the Partner Public Profile page, Partner Landing page, Partner Registration, Customer Snapshot, and Customer Overview - Analysis tab), and customize transactional pages (all other non-dashboard pages).

<table>
<thead>
<tr>
<th>Use Cases</th>
<th>Design View - Standard mode</th>
<th>Design View - Direct Selection mode</th>
<th>Page Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add content (Business Intelligence reports, Sales Cloud portlets such as Calendar)</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Delete region</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Move region</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Change page layout (for example, change a two column layout to three column layout)</td>
<td>Yes</td>
<td>No</td>
<td>Dashboard and other select pages</td>
</tr>
<tr>
<td>Default region state (open or close)</td>
<td>Yes</td>
<td>No</td>
<td>Transactional pages (all non-dashboard pages)</td>
</tr>
<tr>
<td>Manage save queries (create and edit)</td>
<td>Yes</td>
<td>No</td>
<td>Transactional pages (all non-dashboard pages)</td>
</tr>
</tbody>
</table>
Application Composer also lets you make UI changes at run time. However, the types of UI changes that you can make using Application Composer are quite different. Specifically, your primary focus when using Application Composer is to make actual object model changes. For example, you can create a new business object and related fields, and then create new application pages where that object and its fields are exposed to users.

The following table describes some of the primary differences between Page Composer and Application Composer. For example, using Application Composer, you cannot access the Resource Catalog to add new content to a page. With Application Composer, administrators can make customizations at the site level only.
<table>
<thead>
<tr>
<th>Customization Task</th>
<th>Available in Page Composer (site, job role, external or internal level)?</th>
<th>Available in Application Composer (site level only)?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make object model extensions and expose your customizations by creating or modifying work area pages</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Reorder subtabs</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Customize dashboard pages</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Add content from the Resource Catalog</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Simple field customizations (show, hide, make read only, make required)</td>
<td>Yes (WYSIWYG - what you see is what you get)</td>
<td>Yes (non-WYSIWYG)</td>
</tr>
<tr>
<td>Select the MDS layer where you want to author customizations, such as at the site layer or job role layer</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>View results of customizations immediately</td>
<td>Yes, in the Page Composer design interface</td>
<td>Yes, in the Sales Cloud application that you are customizing</td>
</tr>
</tbody>
</table>
Importing and Exporting Setup Data

Configuration Packages: Explained

Almost all Oracle Fusion application implementations require moving functional setup data from one instance into another at various points in the lifecycle of the applications. For example, one of the typical cases in any enterprise application implementation is to first implement in a development or test application instance and then deploy to a production application instance after thorough testing. You can move functional setup configurations of applications from one application instance into another by exporting and importing Configuration packages from the Manage Configuration Packages page.

A Configuration Package contains the setup import and export definition. The setup import and export definition is the list of setup tasks and their associated business objects that identifies the setup data for export as well as the data itself. When you create a configuration package only the setup export and import definition exists. Once you export the configuration package appropriate setup data is added to the configuration package using the definition. Once a configuration package is exported, the setup export and import definition is locked and cannot be changed.

You generate the setup export and import definition by selecting an implementation project and creating a configuration package. The tasks and their associated business objects in the selected implementation project define the setup export and import definition for the configuration package. In addition, the sequence of the tasks in the implementation project determine the export and import sequence.

Exporting and Importing Setup Data: Explained

A configuration package is required to export setup data. You can export a configuration package once you create it, or at any time in the future. During export, appropriate setup data will be identified based on the setup export definition and added to the configuration package. The setup data in the configuration package is a snapshot of the data in the source application instance at the time of export. After the export completes, you can download the configuration package as a zipped archive of multiple XML files, move it to the target application instance, and upload and import it.
Export
You can export a configuration package multiple times by creating multiple versions. While the export definition remains the same in each version, the setup data can be different if you modified the data in the time period between the different runs of the export process. Since each version of the configuration package has a snapshot of the data in the source instance, you can compare and analyze various versions of the configuration package to see how the setup data changed.

Import
In the target application instance, the setup import process will insert all new data from the source configuration package that does not already exist and update any existing data with changes from the source. Setup data that exists in the target instance but not in source will remain unchanged.

Export and Import Reports
You can review the results of the export and import processes using reports. The results appear ordered by business objects and include information on any errors encountered during the export or import process. If a setup export or import process paused due to errors encountered or for a manual task to be performed outside of the application, then you can resume the paused process.

These reports show what setup data was exported or imported and by which specific process. You can change the reports to validate the setup data as well as to compare or analyze it. A report is generated for each business object. These reports show the same information as the export and import results seen directly in the application.

Process status details are available as text files showing the status of an export or import process including the errors encountered during the process.

Moving Common Reference Objects

Moving Common Reference Objects: Overview

The common reference objects in Oracle Middleware Extensions for Applications are used by several setup tasks in the Setup and Maintenance work area. The common reference objects become a part of the configuration package that is created for an implementation project. While moving the application content, for example, from the test phase to the production phase of an implementation, you must pay special attention to the nuances of these common reference objects.

Parameters
The common reference objects are represented as business objects. A single object can be referenced in multiple setup tasks with different parameters. In the configuration package that is created for the implementation project, parameters passed to a setup task are also passed to the business objects being moved. As a result, the scope of the setup tasks is maintained intact during the movement.
Dependences
Common reference objects may have internal references or dependencies among other common reference objects. Therefore, it is necessary that all the dependencies are noted before the movement of objects so that there are no broken references among the objects.

Business Objects for Moving Common Reference Objects: Points to Consider

Common reference objects in Oracle Fusion Functional Setup Manager are represented by business objects. These business objects are the agents that contain the application content and carry them across whenever the application setup is moved from one environment to another, for example, test environment to production environment.

Choice of Parameters
The following table lists the business objects, the corresponding movement details, and the effect of the setup task parameter on the scope of the movement.

Note
- Only the translation in the current user language is moved.
- The Oracle Social Network business objects and the Navigator menu customizations are moved using the customization sets on the Customization Migration page instead of using the export and import function in the Setup and Maintenance work area.

<table>
<thead>
<tr>
<th>Business Object Name</th>
<th>Moved Functional Item</th>
<th>Effect on the Scope of Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Message</td>
<td>Messages and associated tokens</td>
<td>No parameters: all messages are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>messageName/applicationId only the specified message is moved.</td>
</tr>
<tr>
<td>Application Taxonomy</td>
<td>Application taxonomy modules and components</td>
<td>No parameters: all taxonomy modules and components are moved.</td>
</tr>
<tr>
<td>Application Attachment Entity</td>
<td>Attachment entities</td>
<td>No parameters: all attachment entities are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey only attachment entities belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Attachment Category</td>
<td>Attachment categories and category-to-entity mappings</td>
<td>No parameters: all attachment categories and category-to-entity mappings are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey only attachment categories belonging to the specified module and its descendant modules in the taxonomy hierarchy along with the respective category-to-entity mappings are moved.</td>
</tr>
<tr>
<td>Application Document Sequence Category</td>
<td>Document sequence categories</td>
<td>No parameters: all categories are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>code/applicationId only the specified document sequence category code is moved.</td>
</tr>
<tr>
<td>Application Document Sequence</td>
<td>Document sequences and their assignments</td>
<td>No parameters: all sequences are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey only document sequences belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved</td>
</tr>
<tr>
<td></td>
<td></td>
<td>name: only the specified document sequence is moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 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.  
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.  
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.  
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.  
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved. |
<table>
<thead>
<tr>
<th>Business Object Name</th>
<th>Moved Functional Item</th>
<th>Effect on the Scope of Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Extensible Flexfield</td>
<td>Extensible flexfield registration data and setup data, including categories</td>
<td>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. 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. Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Key Flexfield</td>
<td>Key flexfield registration data and setup data</td>
<td>No parameters: all key flexfields are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only key flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>keyFlexfieldCode/ applicationId only the specified key flexfield is moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Importing a flexfield's metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically</td>
</tr>
<tr>
<td></td>
<td></td>
<td>submits affected flexfields for redeployment.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Importing a flexfield's metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically</td>
</tr>
<tr>
<td></td>
<td></td>
<td>submits affected flexfields for redeployment.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Flexfield Value Set</td>
<td>Value set setup data</td>
<td>No parameters: all value sets are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only value sets belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>valueSetCode: only the specified value set is moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Importing a value set’s metadata can change the deployment status of flexfields that use the value set, and therefore the affected flex fields must be redeployed. The import process automatically submits affected flexfields for redeployment.</td>
</tr>
<tr>
<td>Application Reference Currency</td>
<td>Currency data</td>
<td>No parameters: all currencies are moved.</td>
</tr>
<tr>
<td>Application Reference ISO Language</td>
<td>ISO language data</td>
<td>No parameters: all ISO languages are moved.</td>
</tr>
<tr>
<td>Application Reference Industry</td>
<td>Industry data including industries in territories data</td>
<td>No parameters: all industries are moved.</td>
</tr>
<tr>
<td>Application Reference Language</td>
<td>Language data</td>
<td>No parameters: all languages are moved.</td>
</tr>
<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>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Standard Lookup</td>
<td>Standard lookup types and their lookup codes</td>
<td>No parameters: all standard lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Common Lookup</td>
<td>Common lookup types and their lookup codes</td>
<td>No parameters: all common lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey - only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Set-Enabled Lookup</td>
<td>Set-enabled lookup types and their lookup codes</td>
<td>No parameters: all set-enabled lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified set-enabled lookup is moved.</td>
</tr>
<tr>
<td>Application Profile Category</td>
<td>Profile categories</td>
<td>No parameters: all profile categories are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>name/ applicationId only the specified category is moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Profile Option</td>
<td>Profile options and their values</td>
<td>No parameters: all profile options and their values are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only profile options and their values belonging to the specified module are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>profileOptionName: only the specified profile option and its values are moved.</td>
</tr>
<tr>
<td>Application Profile Value</td>
<td>Profile options and their values</td>
<td>No parameters: all profiles and their values are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only profiles and their values belonging to the specified module are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>categoryName/ categoryApplicationId only profiles and their values belonging to the specified category are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>profileOptionName: only the specified profile and its values are moved.</td>
</tr>
<tr>
<td>Application Reference Data Set</td>
<td>Reference data sets</td>
<td>No parameters: all sets are moved.</td>
</tr>
<tr>
<td>Application Reference Data Set</td>
<td>Reference data set assignments</td>
<td>determinantType: only assignments for the specified determinant type are moved.</td>
</tr>
<tr>
<td>Assignment</td>
<td></td>
<td>determinantType/ referenceGroupName only assignments for the specified determinant type and reference group are moved.</td>
</tr>
<tr>
<td>Application Tree Structure</td>
<td>Tree structures and any labels assigned to the tree structure</td>
<td>No parameters: all tree structures (and their labels) are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only tree structures (and their labels) belonging to the specified module are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>treeStructureCode: only the specified tree structure (with its labels) is moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Tree</td>
<td>Tree codes and versions</td>
<td>No parameters: all trees are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only trees belonging to the specified module are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>treeStructureCode: only trees belonging to the specified tree structure are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TreeStructureCode/ TreeCode only trees belonging to the specified tree structure and tree code are moved.</td>
</tr>
<tr>
<td>Application Tree Label</td>
<td>Tree structures and any labels assigned to the tree structure</td>
<td>No parameters: all tree structures (and their labels) are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only tree structures (and their labels) belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>treeStructureCode: only the specified tree structure (with its labels) is moved.</td>
</tr>
<tr>
<td>Business Object Name</td>
<td>Moved Functional Item</td>
<td>Effect on the Scope of Movement</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application Data Security Policy</td>
<td>Database resources, actions, conditions, and data security policies</td>
<td>No parameters: all database resources/ actions/ conditions/ policies are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/ moduleKey only database resources/ actions/ conditions/ policies belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>objName: only the specified database resource along with its actions/ conditions/ policies is moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the policies being moved contain reference to newly created roles, move the roles before moving the policies.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the source and target systems use different LDAPs, manually perform the GUID reconciliation after moving the data security policies.</td>
</tr>
<tr>
<td>Application Activity Stream Configuration</td>
<td>Activity stream options</td>
<td>No parameters: all activity stream options are moved.</td>
</tr>
</tbody>
</table>

**Moving Related Common Reference Objects: Points to Consider**

Certain common reference objects may use other common reference objects creating dependencies among the objects. During the movement of common reference objects, these dependencies or references need to be taken care of.

**Dependencies**

The dependencies among the common reference objects may be caused by any of the following conditions.

- Flexfield segments use value sets
- Value sets may make use of standard, common, or set-enabled lookups
• Key flexfields may have an associated tree structure and key flexfield segments may have an associated tree code
• Tree codes and versions may be defined over values of a value set
• Data security policies may be defined for value sets that have been enabled for data security

You may choose to move one, some, or all of the business objects by including the ones you want to move in your configuration package. For example, you may choose to move only value sets and not lookups, or you may choose to move both value sets and their lookups as part of the same package. Whatever be the combination, it is recommended that during the movement of objects, you follow an order that maintains the dependencies among the objects.

While moving the business objects, adhere to the guidelines and exactly follow the order as listed below.

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.

**Note**
In scenarios where there may be dependencies on other objects, you must move the dependencies before moving the referencing object. For example, if data security policies being moved have dependencies on newly created security roles, you must move the security roles before moving the security policies.

**Using Seed Data Framework to Move Common Reference Objects: Points to Consider**

To move the common reference objects, you can use the Seed Data Framework (SDF). You can also use the command line interface of SDF to move the object setup data. For more information about seed data loaders including common reference object loaders, see Oracle Fusion Applications Developer’s Guide.

**Movement Dependencies**

The seed data interface moves only the setup metadata. For example, if you use SDF to import flexfield metadata, the flexfield setup metadata is imported into
your database. However, you must invoke the flexfield deployment process separately after seed data import to regenerate the runtime flexfield artifacts in the target environment. Similarly, if you use SDF to import data security metadata, you must first move any new referenced roles and then manually run the GUID reconciliation where required.

To ensure that the reference data is not lost during the movement, certain guidelines are prescribed. It is recommended that you perform the movement of object data exactly in the order given below.

---

**Note**

Only the translation in the current user language is moved.

---

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.
8. Move created reference data sets before moving any objects that reference them, such as reference data set assignments and set-enabled lookups.
9. Move created document sequence categories before moving any objects that reference them, such as document sequences.
10. Move created tree labels before moving any objects that reference them, such as trees.
11. Move created data security objects and policies before moving any objects that reference them, such as value sets.
12. Move created value sets before moving any objects that reference them, such as flexfields.
13. Move created trees before moving any objects that reference them, such as key flexfields.
abstract role
A description of a person’s function in the enterprise that is unrelated to the person’s job (position), such as employee, contingent worker, or line manager. A type of enterprise role.

accounting flexfield
The chart of accounts that determines the structure, such as the number and order of individual segments, as well as the corresponding values per segment.

action
The kind of access named in a security policy, such as view or edit.

address style format
Specifies the layout of an address, such as how many address lines it contains, and whether a city name is mandatory.

ADF
Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

application identity
Predefined application level user with elevated privileges. An application identity authorizes jobs and transactions for which other users are not authorized, such as a payroll run authorized to access a taxpayer ID while the user who initiated the job is not authorized to access such personally identifiable information.

application role
A role specific to applications and stored in the policy store.

Applications Core
Abbreviation for Oracle Middleware Extensions for Applications. The technical product code is FND.

assignment
A set of information, including job, position, pay, compensation, managers, working hours, and work location, that defines a worker’s or nonworker’s role in a legal employer.

balancing segment
A chart of accounts segment used to automatically balance all journal entries for each value of this segment.
beneficiary
A person or organization designated to receive benefits from a compensation plan on the death of the plan participant.

BPEL
Business Process Execution Language; a standard language for defining how to send XML messages to remote services, manipulate XML data structures, receive XML messages asynchronously from remote services, manage events and exceptions, define parallel sequences of execution, and undo parts of processes when exceptions occur.

business function
A business process, or an activity that can be performed by people working within a business unit and describes how a business unit is used.

business object
A resource in an enterprise database, such as an invoice or purchase order.

business unit
A unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy.

calendar event
A period that signifies an event, such as a public holiday or a training course, that impacts worker availability.

chart of accounts
The account structure your organization uses to record transactions and maintain account balances.

classification category
A grouping of classification codes, categorizing entities such as parties, tasks, and orders. Allows classification code assignment rules to be defined.

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.
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 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 built using Contract Expert.

corporate rate type
Rate you define to standardize rates used in conversion of one currency to another over a period of time. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.

cost center
A unit of activity or group of employees used to assign costs for accounting purposes.

cost organization
A grouping of inventory organizations that indicates legal and financial ownership of inventory, and which establishes common costing and accounting policies.
country holding company
A legal entity that acts on behalf of several divisions within an enterprise, and is the legal employer in a country.

data dimension
A stripe of data accessed by a data role, such as the data controlled by a business unit.

data instance set
The set of human capital management (HCM) data, such as one or more persons, organizations, or payrolls, identified by an HCM security profile.

data role
A role for a defined set of data describing the job a user does within that defined set of data. A data role inherits job or abstract roles and grants entitlement to access data within a specific dimension of data based on data security policies. A type of enterprise role.

data role template
A template used to generate data roles by specifying which base roles to combine with which dimension values for a set of data security policies.

data security
The control of access to data. Data security controls what action a user can taken against which data.

data security policy
A grant of entitlement to a role on an object or attribute group for a given condition.

database resource
An applications data object at the instance, instance set, or global level, which is secured by data security policies.

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.
**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 expiry. A document type exists for a combination of document category and subcategory.

**duty role**  
A group of function and data privileges representing one duty of a job. Duty roles are specific to applications, stored in the policy store, and shared within an Oracle Fusion Applications instance.

**employment terms**  
A set of information about a nonworker's or employee's job, position, pay, compensation, working hours, and work location that all assignments associated with the employment terms inherit.
**enterprise**
An organization with one or more legal entities under common control.

**enterprise role**
Abstract, job, and data roles are shared across the enterprise. An enterprise role is an LDAP group. An enterprise role is propagated and synchronized across Oracle Fusion Middleware, where it is considered to be an external role or role not specifically defined within applications.

**entitlement**
Grants of access to functions and data. Oracle Fusion Middleware term for privilege.

**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.

grade
A component of the employment model that defines the level of compensation for a worker.

HCM data role
A job role, such as benefits administrator, associated with specified instances of Oracle Fusion Human Capital Management (HCM) data, such as one or more positions or all persons in a department.

hierarchy version
Hierarchy version refers to a specific version of a hierarchy. You can create any number of versions, but only one hierarchy version can be active on a specific date. In tree terminology, hierarchy versions are also called tree versions.

identifier type
An identifier type is a specific category of additional identifiers, numeric or otherwise, which can identify members. Examples of identifier types could be bank account numbers, passport numbers, and so on.

identity
A person representing a worker, supplier, or customer.

interface table
A database table used for transferring data between applications or from an external application or data file.

inventory organization
An organization that tracks inventory transactions and balances, and can manufacture or distribute products.

item master
A collection of data that describes items and their attributes recorded in a database file.

Items
Entries within the Product master database. For example, items for a manufacturing company can include nuts, bolts, and screws.

job
A generic role that is independent of any single department or location. For example, the jobs Manager and Consultant can occur in many departments.
**job role**

A role for a specific job consisting of duties, such as an accounts payable manager or application implementation consultant. A type of enterprise role.

**key flexfield**

Configurable key consisting of multiple parts or segments, each of which may be meaningful individually or in combination with the others. Key flexfields are commonly implemented to represent part numbers and account numbers.

**key flexfield segment instance**

A single occurrence of a key flexfield segment in a key flexfield structure instance.

**key flexfield structure**

The arrangement of segments in a key flexfield. In some cases, multiple structures can be defined for a single key flexfield.

**key flexfield structure instance**

A single occurrence of a key flexfield structure that shares the same order of segments as every other instance of the key flexfield structure, but uses different value sets to validate the segments.

**legal employer**

A legal entity that employs people.

**legal entity**

An entity is identified and given rights and responsibilities under commercial law, through the registration with the country’s appropriate authority.

**legal reporting unit**

The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.

**legislative data group**

A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.

**line of business**

Set of one or more highly related products which service a particular customer transaction or business need. Refers to an internal corporate business unit.
lookup code
A value available for lookup within a lookup type such as the code BLUE within the lookup type COLORS.

lookup type
A set of lookup codes to be used together as a list of values on a field in the user interface.

mainline
A branch of data that serves as a single source of truth.

managed person
In Oracle Fusion Human Capital Management security, a person for whom the user can maintain some information. For example, line managers can maintain information about their direct and indirect reports, and workers can maintain information about themselves, their dependents, and their beneficiaries.

marketing list
A static selection of contacts for the purpose of communicating a marketing message by e-mail, direct mail or phone.

model profile
A collection of the work requirements and required skills and qualifications of a workforce structure, such as a job or position.

name style format
Specifies the layout of a name, such as first name, last name, and phonetic last name.

name type
A name type is a specific category of additional names that can identify trading community members. Examples of name types could be maiden names, aliases, doing-business-as names, and so on.

natural account
Categorizes account segment values by account type, asset, liability, expense, revenue, or equity, and sets posting, budgeting, and other options.

offering
A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.
organization hierarchy
A tree structure that determines the relationships between organizations, such as which organizations are subordinate to other organizations.

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.

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 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.

**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.

**resource**
People designated as able to be assigned to work objects, for example, service agents, sales managers, or partner contacts. A sales manager and partner contact can be assigned to work on a lead or opportunity. A service agent can be assigned to a service request.

**resource organization**
An organization whose members are resources. Resource organizations are used to implement sales organizations, partner organizations, and so on.

**resource role**
Resource roles indicate the role a resource plays as an individual, or within a resource team.

**resource team**
A resource team is a temporary group of resources formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization.

**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.

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.

survivorship
The process of creating the best version record intelligently, especially from multiple source systems, by specifying criteria for selecting record and attributes to be retained during merge or link, or update operations.

system person type
A fixed name that the application uses to identify a group of people.

territory
A legally distinct region that is used in the country field of an address.

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.

value set
A set of valid values against which values entered by an end user are validated. The set may be tree structured (hierarchical).

work relationship
An association between a person and a legal employer, where the worker type determines whether the relationship is a nonworker, contingent worker, or employee work relationship.
**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.